Ch.6 – OSPF
Part 1 of 2: Single Area OSPF

Cabrillo College

CCNP 1 version 3.0

Rick Graziani

Cabrillo College

Note to instructors

- If you have downloaded this presentation from the Cisco Networking Academy Community FTP Center, this may not be my latest version of this PowerPoint.
- For the latest PowerPoints for all my CCNA, CCNP, and Wireless classes, please go to my web site:

  http://www.cabrillo.edu/~rgraziani/

  - The username is *cisco* and the password is *perlman* for all of my materials.
- If you have any questions on any of my materials or the curriculum, please feel free to email me at graziani@cabrillo.edu (I really don't mind helping.) Also, if you run across any typos or errors in my presentations, please let me know.
- I will add "(Updated – *date*)" next to each presentation on my web site that has been updated since these have been uploaded to the FTP center.

Thanks! Rick

Notes

Upon completion of this module, the student will be able to perform tasks related to the following:	
6.1	OSPF Overview
6.2	OSPF Operation
6.3	OSPF Configuration and Verification
6.4	Configuring OSPF Over NMBA
6.5	Multiarea OSPF Operation
6.6	Multiarea OSPF Configuration and Verification
6.7	Stub, Totally Stubby, and Not-So-Stubby Areas
6.8	Virtual Links
6.9	OSPF Verification and Configuration Lab Exercises
6.10	Creating Multiarea OSPF Challenge Lab

- Configuration of OSPF is easy.
- The concepts and theory that make it a robust and scalable protocol is a little more complex.
- Information in this presentation that goes beyond that which is presented in the CCNP 3.0 curriculum.
- This information is included to give you a better understanding of OSPF, to answer some of the students' questions, and to get an idea of the true operational features of OSPF.

Preview of the OSPF Commands

Required Commands:

```
Rtr(config)# router ospf process-id
Rtr(config-router)#network address wildcard-mask area area-id
```

Preview of the OSPF Commands

Optional Commands:

```
Rtr(config-router)# default-information originate (Send default)
Rtr(config-router)# area area authentication  (Plain authen.)
Rtr(config-router)# area area authentication message-digest
                                               (md5 authen.)
Rtr(config)# interface loopback number     (Configure lo as RtrID)
Rtr(config)# interface type slot/port
Rtr(config-if)# ip ospf priority <0-255>     (DR/BDR election)
Rtr(config-if)# bandwidth kbps           (Modify default bandwdth)
RTB(config-if)# ip ospf cost cost        (Modify inter. cost)
Rtr(config-if)# ip ospf hello-interval seconds     (Modify Hello)
Rtr(config-if)# ip ospf dead-interval seconds      (Modify Dead)
Rtr(config-if)# ip ospf authentication-key passwd (Plain/md5authen)
Rtr(config-if)# ip ospf message-digest-key key-id md5 password
```

Introduction to OSPF Concepts

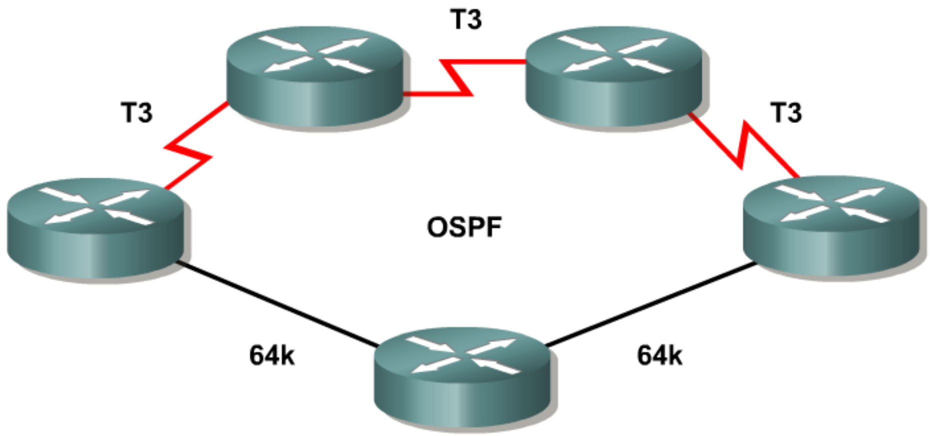
Introducing OSPF and Link State Concepts

- **Advantages of OSPF**
- **Brief History**
- **Terminology**
- **Link State Concepts**

Introducing the OSPF Routing Protocol

- **Metric based on Cost (Bandwidth)**
- **Hello Protocol**
- **Steps to OSPF Operation**
- **DR/BDR**
- **OSPF Network Types**

Advantages of OSPF (1 of 2)



- **OSPF is link-state routing protocol**
  - RIP, IGRP and EIGRP are distance-vector (routing by rumor) routing protocols, susceptible to routing loops, split-horizon, and other issues.
- **OSPF has fast convergence**
  - RIP and IGRP hold-down timers can cause slow convergence.
- **OSPF supports VLSM and CIDR**
  - RIPv1 and IGRP do not

Advantages of OSPF (2 of 2)

- **Cisco's OSPF metric is based on bandwidth**
  - RIP is based on hop count
  - IGRP/EIGRP bandwidth, delay, reliability, load
- **OSPF only sends out changes when they occur**.
  - RIP sends entire routing table every 30 seconds, IGRP every 90 seconds
  - Extra: With OSPF, a router does flood its own LSAs when it age reaches 30 minutes (later)
- **OSPF also uses the concept of areas to implement hierarchical routing**
- **Two open-standard routing protocols to choose from:**
  - RIP, simple but very limited, or
  - OSPF, robust but more sophisticated to implement.
- **IGRP and EIGRP are Cisco proprietary**

Brief History

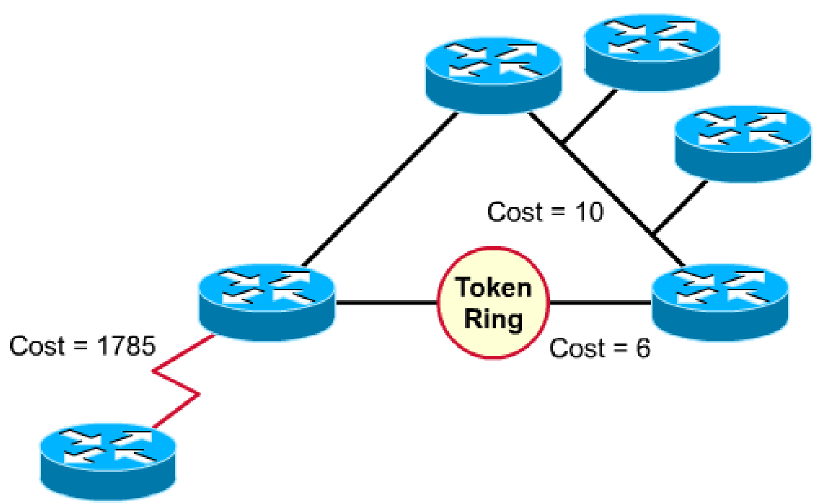
Cabrillo College

- The first link-state routing protocol was implemented and deployed in the ARPANET (Advanced Research Project Agency Network), the predecessor to later link-state routing protocols.
- Next, DEC (Digital Equipment Corporation) proposed and designed a link-state routing protocol for ISO's OSI networks, IS-IS (Intermediate System-to-Intermediate System).
  - Later, IS-IS was extended by the IETF to carry IP routing information.
- **IETF working group** designed a routing protocol specifically for IP routing, OSPF (Open Shortest Path First).
- OSPF version 2, current version, **RFC 2328, John Moy**
- Uses the **Dijkstra algorithm** to calculate a SPT *(Shortest Path Tree)*

Terminology

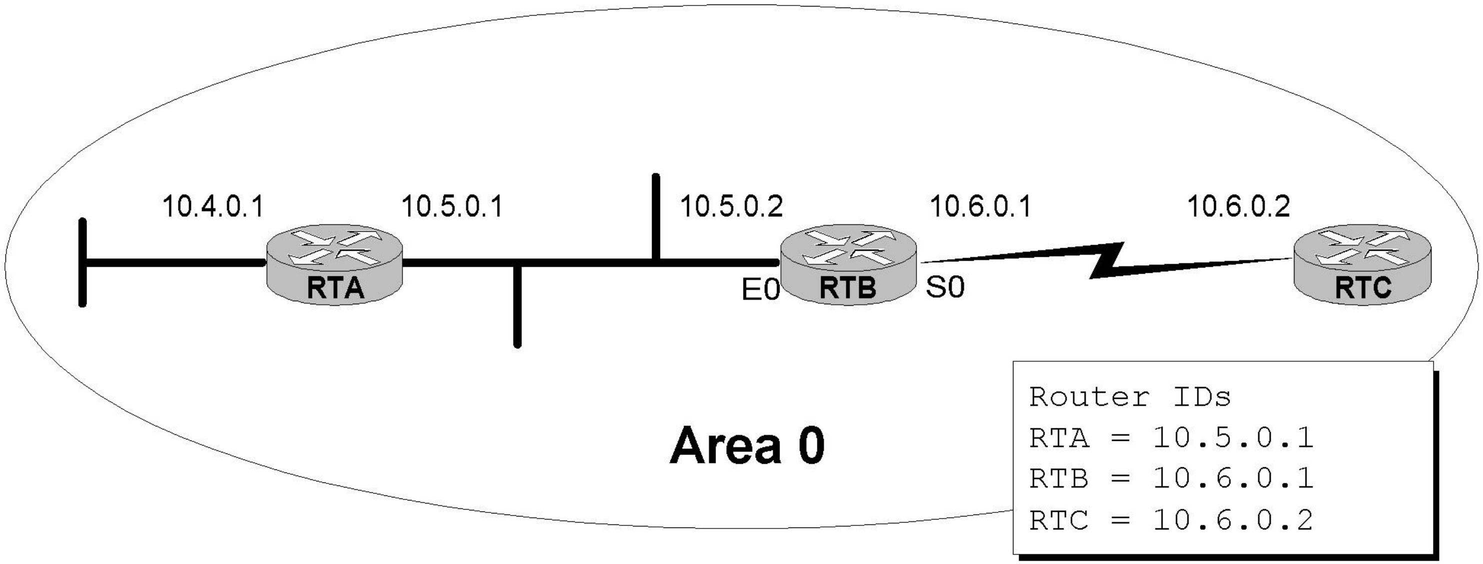
Cabrillo College

- **Link**: Interface on a router
- **Link state**: Description of an interface and of its relationship to its neighboring routers, including:
  - IP address/mask of the interface,
  - The type of network it is connected to
  - The routers connected to that network
  - The metric (cost) of that link
- The collection of all the link-states would form a **link-state database**.



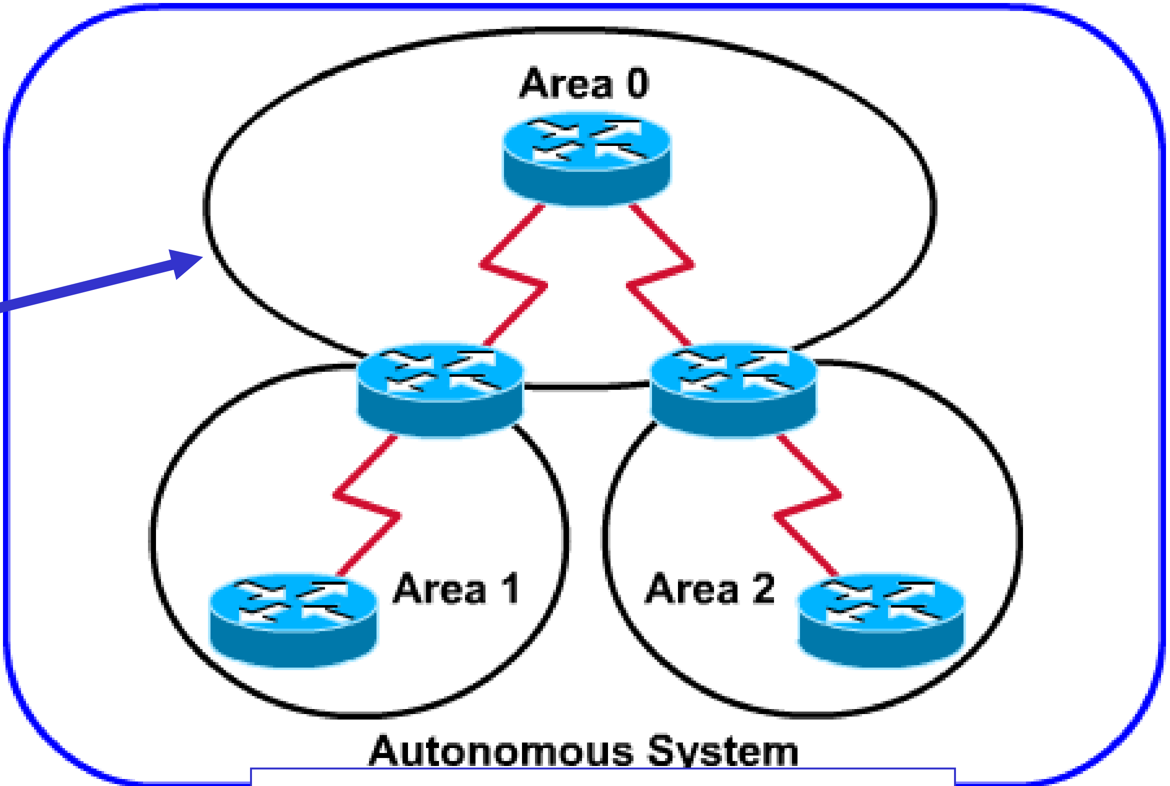
Terminology

Cabrillo College



- **Router ID** – Used to identify the routers in the OSPF network
  - – IP address configured with the OSPF **`router-id`** command (extra)
  - – **Highest loopback address** (configuration coming)
  - – **Highest active IP address** (any IP address)
- Loopback address has the advantage of never going down, thus diminishing the possibility of having to re-establish adjacencies. (more in a moment)

Terminology



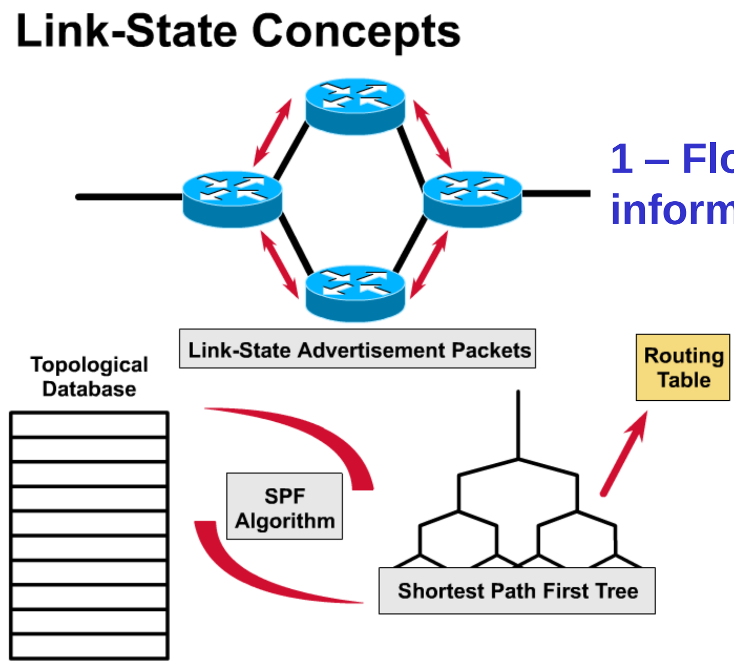
Single Area OSPF uses only one area, usually Area 0

Or "OSPF Routing Domain"

CCNA 3.0 covers Single Area OSPF as opposed to Multi-Area OSPF

- All routers will be configured in a single area, the convention is to use area 0
- If OSPF has more than one area, it must have an area 0
- CCNP includes Multi-Area OSPF
- We will include a brief introduction to Multi-Area OSPF so you can see the real advantages to using OSPF

Link State

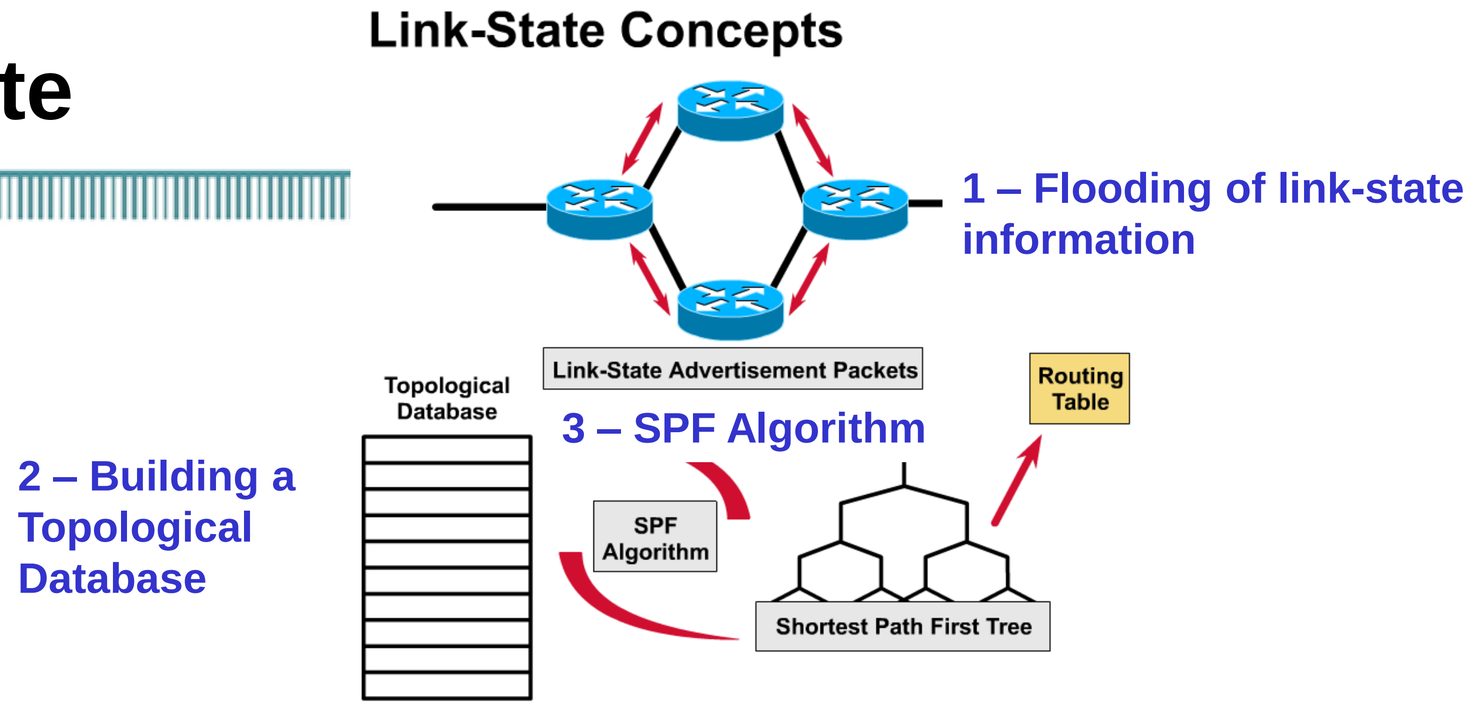


1 – Flooding of link-state information

1 – Flooding of link-state information

- The first thing that happens is that each node, router, on the network announces its own piece of link-state information to other all other routers on the network. This includes who their neighboring routers are and the cost of the link between them.
- Example: “Hi, I’m RouterA, and I can reach RouterB via a T1 link and I can reach RouterC via an Ethernet link.”
- Each router sends these announcements to all of the routers in the network.

Link State



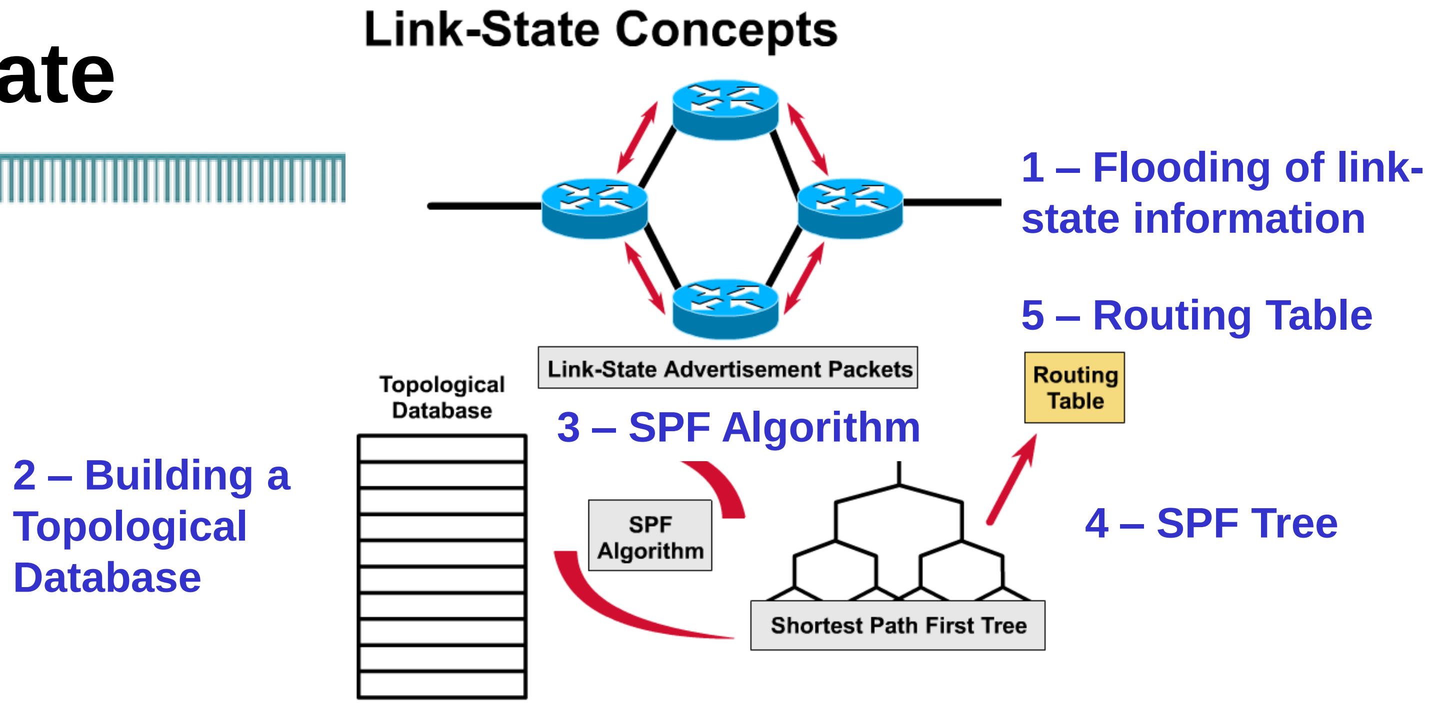
2. Building a Topological Database

- Each router collects all of this link-state information from other routers and puts it into a topological database.

3. Shortest-Path First (SPF), Dijkstra's Algorithm

- Using this information, the routers can recreate a topology graph of the network.
- Believe it or not, this is actually a very simple algorithm and I highly suggest you look at it some time, or even better, take a class on algorithms. (Radia Perlman's book, *Interconnections*, has a very nice example of how to build this graph – she is one of the contributors to the SPF and Spanning-Tree algorithms.)

Link State



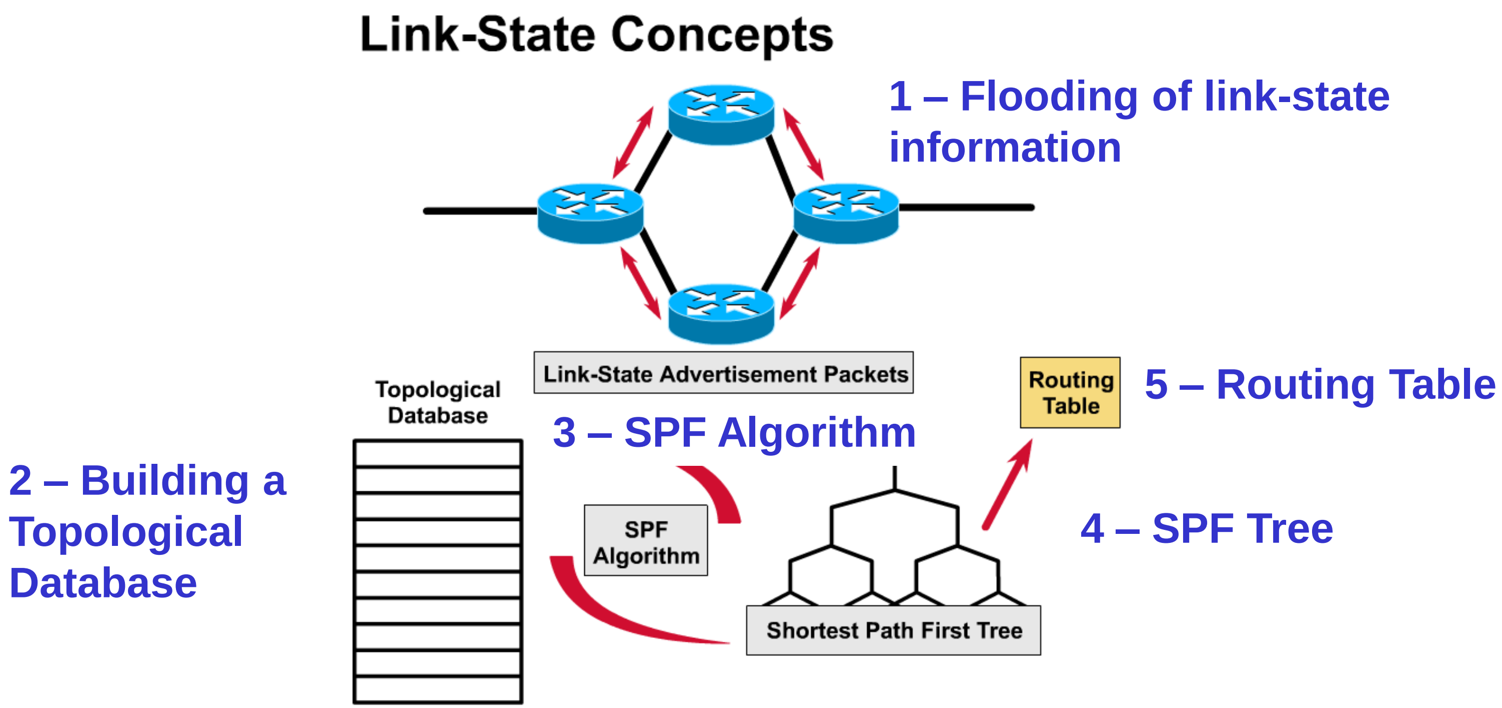
4. Shortest Path First Tree

- This algorithm creates an SPF tree, with the router making itself the root of the tree and the other routers and links to those routers, the various branches.

5. Routing Table

- Using this information, the router creates a routing table.

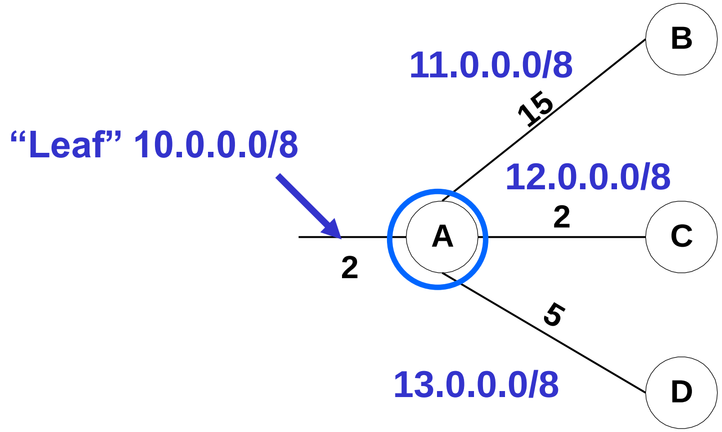
Link State Concepts



- How does the SPF algorithm create an SPF Tree?
- Let's take a look!
- This is *extra* Information.

Extra: Simplified Link State Example

- *In order to keep it simple, we will take some liberties with the actual process and algorithm, but you will get the basic idea!*
- You are **RouterA** and you have exchanged “Hellos” with:
  - RouterB on your network 11.0.0.0/8 with a cost of 15,
  - RouterC on your network 12.0.0.0/8 with a cost of 2
  - RouterD on your network 13.0.0.0/8 with a cost of 5
  - Have a “leaf” network 10.0.0.0/8 with a cost of 2
- This is your link-state information, which you will flood to all other routers.
- All other routers will also flood their link state information. (OSPF: only within the area)



Extra: Simplified Link State Example

RouterA's Topological Data Base (Link State Database)

All other routers flood their own link state information to all other routers.

RouterA gets all of this information and stores it in its LSD (Link State Database).

Using the link state information from each router, RouterC runs Dijkstra algorithm to create a SPT. (next)

RouterB:
- **Connected to RouterA on network 11.0.0.0/8, cost of 15**
- **Connected to RouterE on network 15.0.0.0/8, cost of 2**
- **Has a "leaf" network 14.0.0.0/8, cost of 15**

RouterC:
- **Connected to RouterA on network 12.0.0.0/8, cost of 2**
- **Connected to RouterD on network 16.0.0.0/8, cost of 2**
- **Has a "leaf" network 17.0.0.0/8, cost of 2**

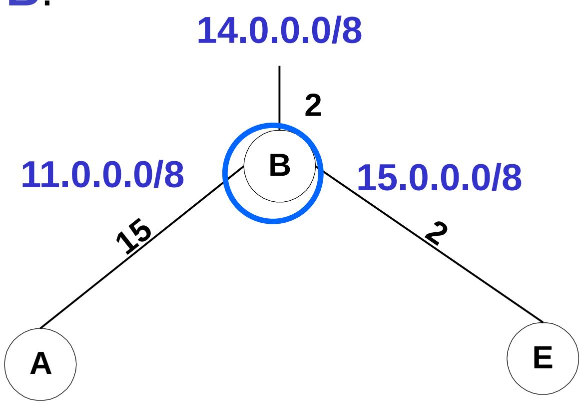
RouterD:
- **Connected to RouterA on network 13.0.0.0/8, cost of 5**
- **Connected to RouterC on network 16.0.0.0/8, cost of 2**
- **Connected to RouterE on network 18.0.0.0/8, cost of 2**
- **Has a "leaf" network 19.0.0.0/8, cost of 2**

RouterE:
- **Connected to RouterB on network 15.0.0.0/8, cost of 2**
- **Connected to RouterD on network 18.0.0.0/8, cost of 10**
- **Has a "leaf" network 20.0.0.0/8, cost of 2**

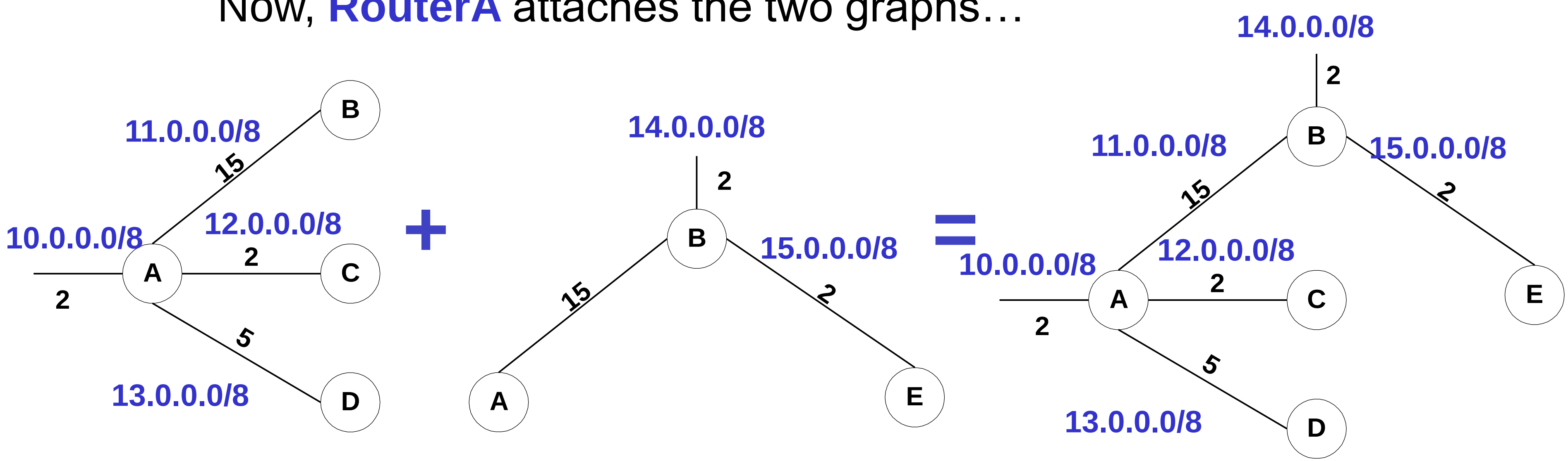
Link State information from RouterB

We now get the following link-state information from **RouterB**:

- Connected to RouterA on network 11.0.0.0/8, cost of 15
- Connected to RouterE on network 15.0.0.0/8, cost of 2
- Have a “leaf” network 14.0.0.0/8, cost of 15



Now, **RouterA** attaches the two graphs…

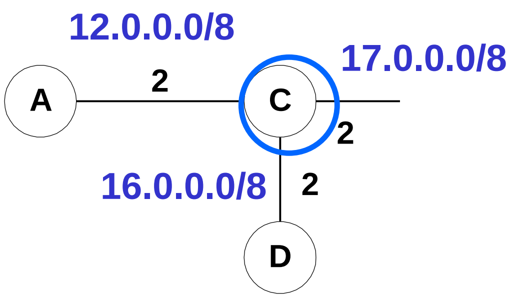


Link State information from RouterC

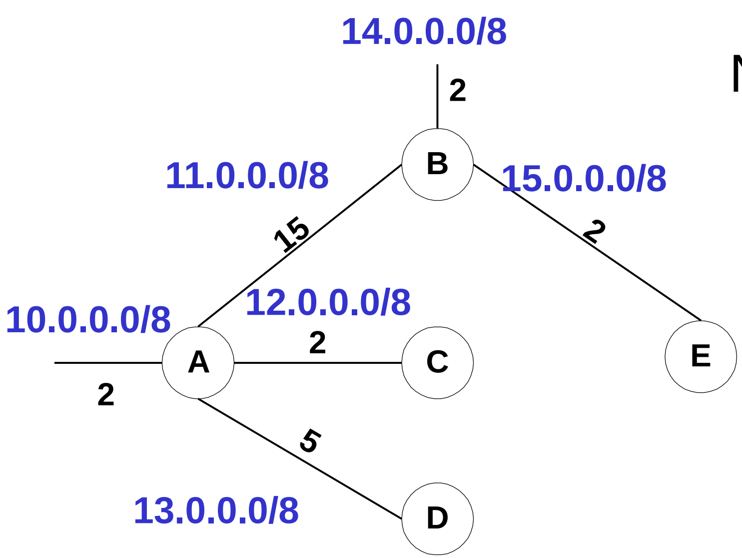
Cabrillo College

We now get the following link-state information from **RouterC**:

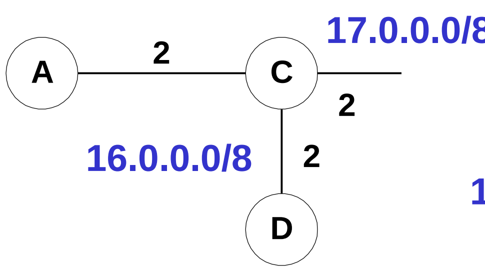
- Connected to RouterA on network 12.0.0.0/8, cost of 2
- Connected to RouterD on network 16.0.0.0/8, cost of 2
- Have a “leaf” network 17.0.0.0/8, cost of 2



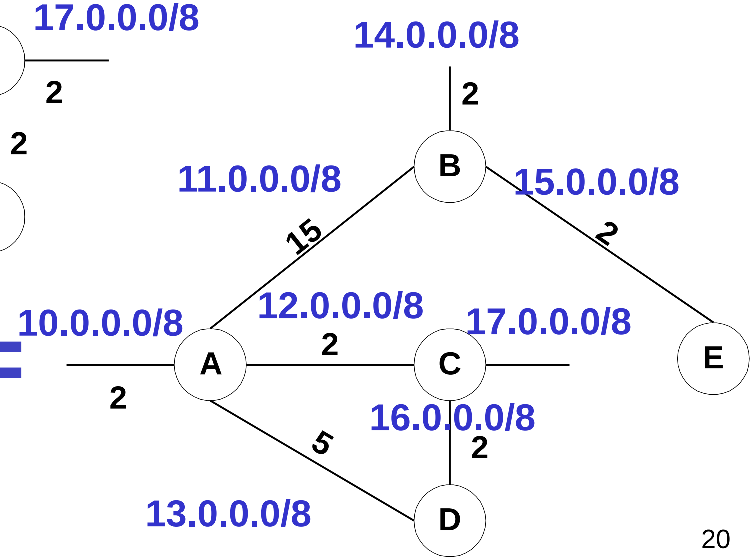
Now, **RouterA** attaches the two graphs…



+



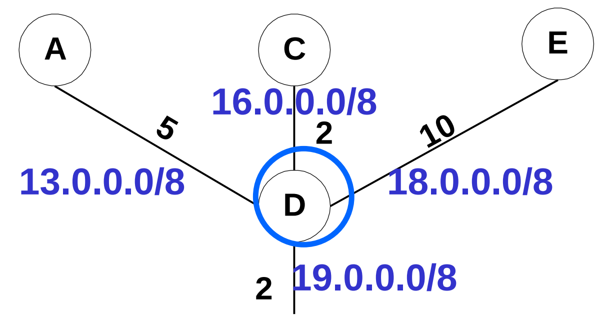
=



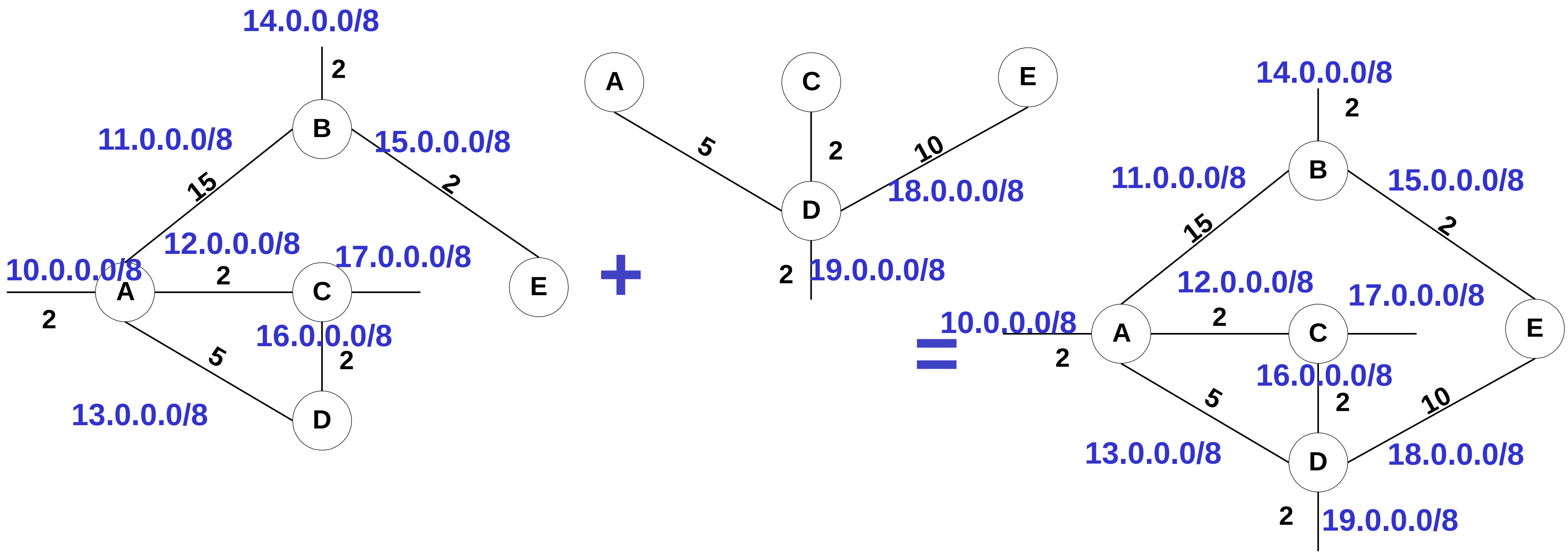
Link State information from RouterD

We now get the following link-state information from **RouterD**:

- Connected to RouterA on network 13.0.0.0/8, cost of 5
- Connected to RouterC on network 16.0.0.0/8, cost of 2
- Connected to RouterE on network 18.0.0.0/8, cost of 2
- Have a “leaf” network 19.0.0.0/8, cost of 2



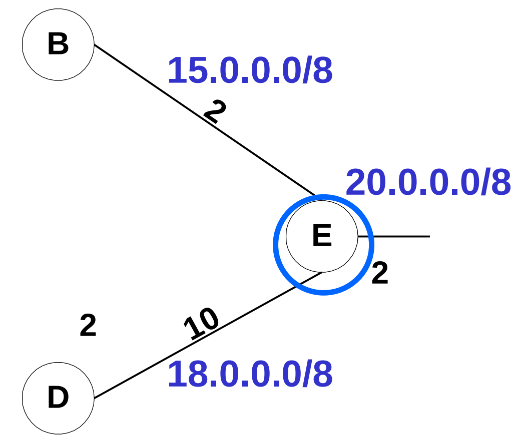
Now, **RouterA** attaches the two graphs…



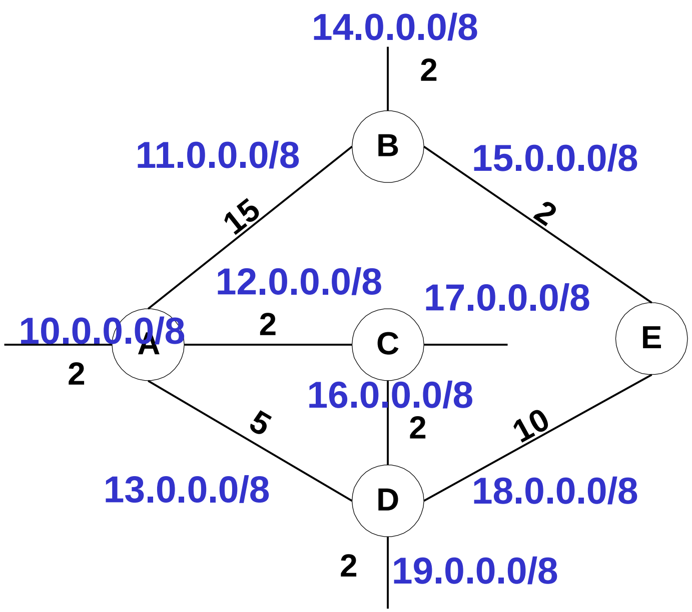
Link State information from RouterE

We now get the following link-state information from **RouterE**:

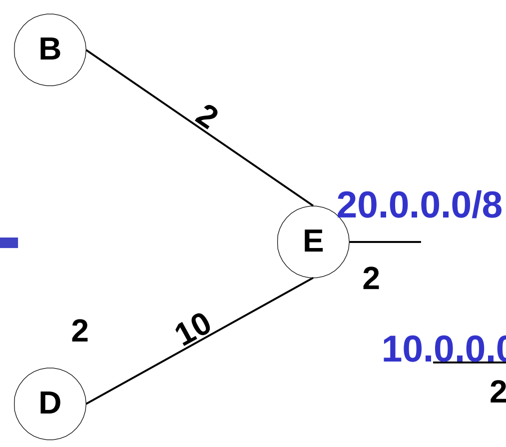
- Connected to RouterB on network 15.0.0.0/8, cost of 2
- Connected to RouterD on network 18.0.0.0/8, cost of 10
- Have a “leaf” network 20.0.0.0/8, cost of 2

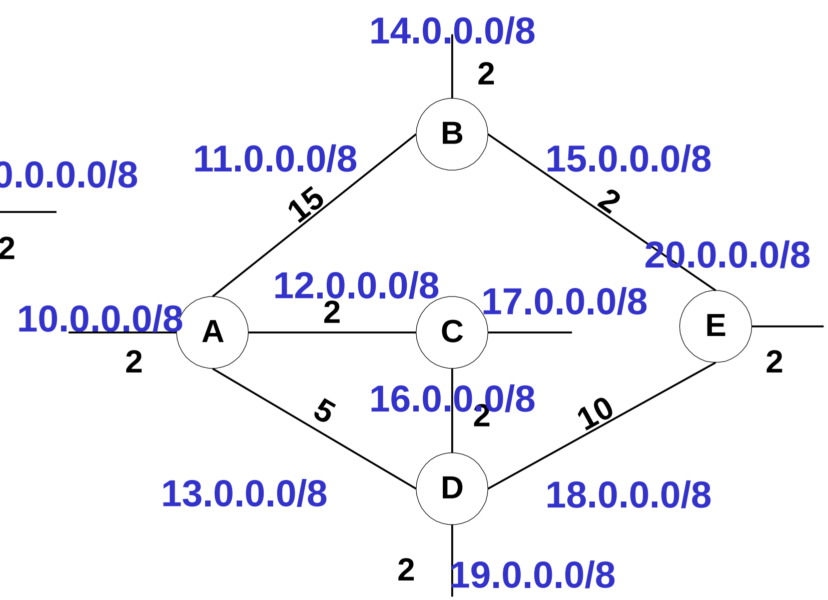


Now, **RouterA** attaches the two graphs…



+

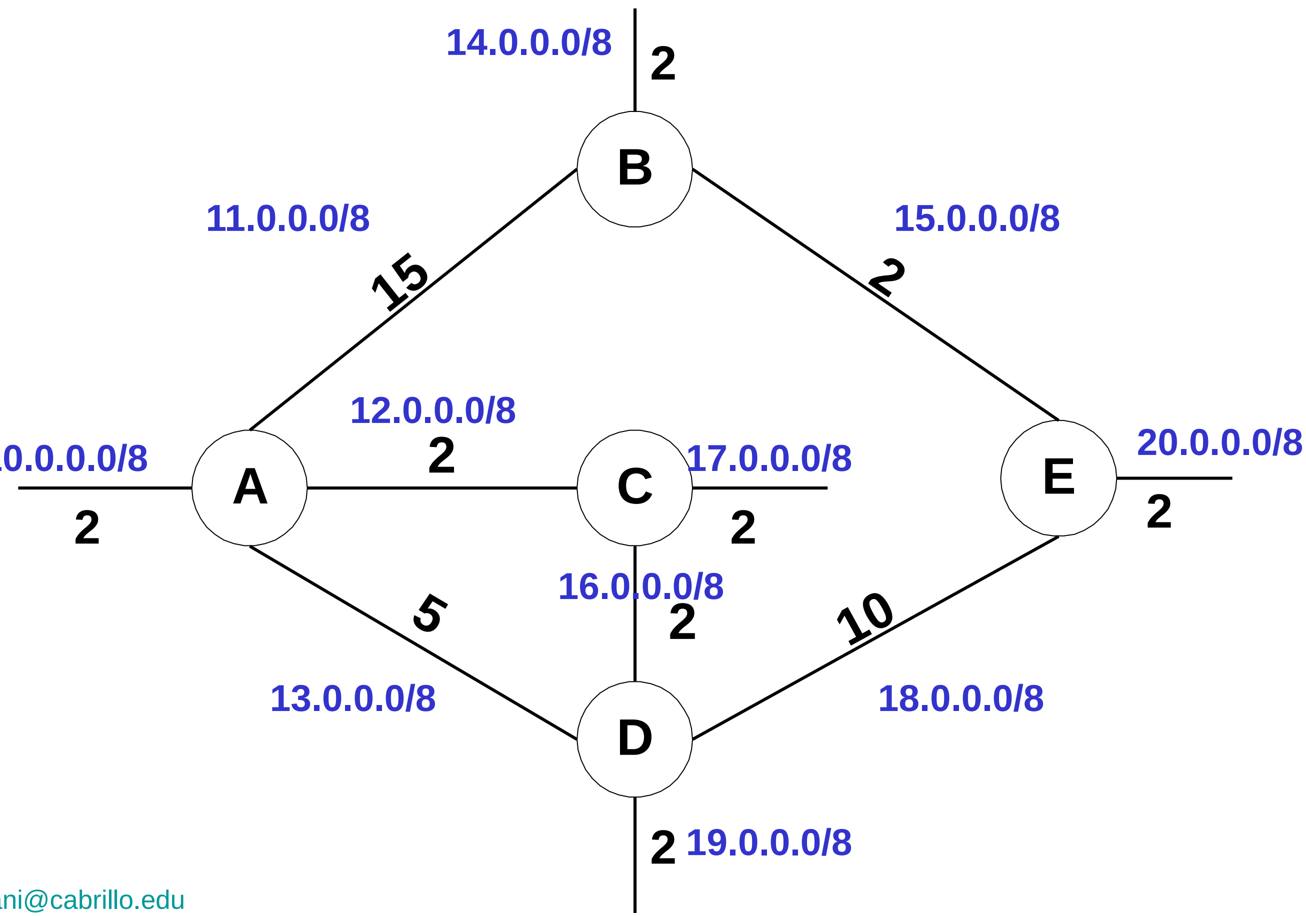




Topology

Cabrillo College

- Using the topological information we listed, RouterA has now built a complete topology of the network.
- The next step is for the link-state algorithm to find the best path to each node and leaf network.



Extra: Simplified Link State Example

RouterA's Topological Data Base (Link State Database)

RouterB:

- **Connected to RouterA on network 11.0.0.0/8, cost of 15**
- **Connected to RouterE on network 15.0.0.0/8, cost of 2**
- **Has a "leaf" network 14.0.0.0/8, cost of 15**

RouterC:

- **Connected to RouterA on network 12.0.0.0/8, cost of 2**
- **Connected to RouterD on network 16.0.0.0/8, cost of 2**
- **Has a "leaf" network 17.0.0.0/8, cost of 2**

RouterD:

- **Connected to RouterA on network 13.0.0.0/8, cost of 5**
- **Connected to RouterC on network 16.0.0.0/8, cost of 2**
- **Connected to RouterE on network 18.0.0.0/8, cost of 2**
- **Has a "leaf" network 19.0.0.0/8, cost of 2**

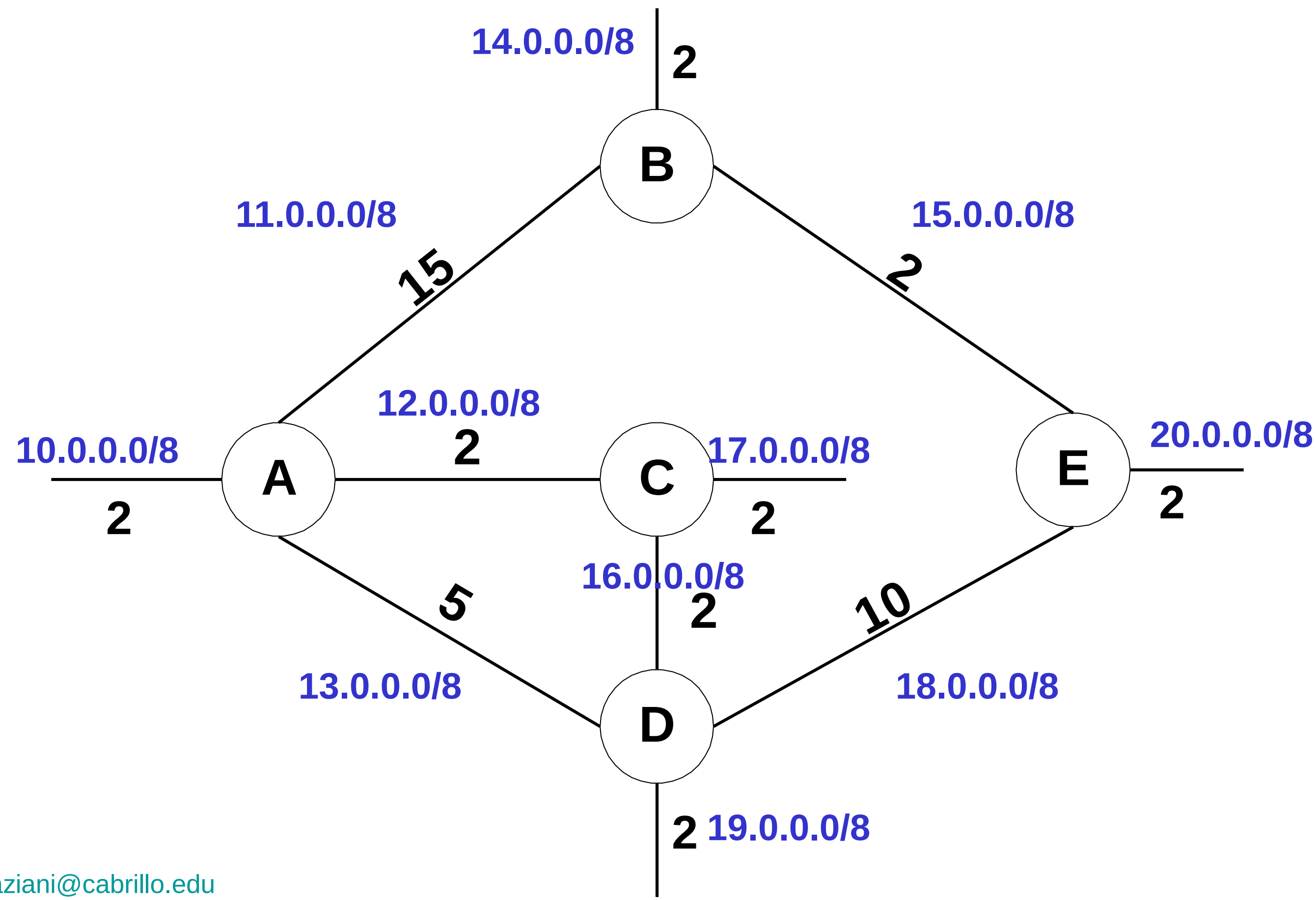
RouterE:

- **Connected to RouterB on network 15.0.0.0/8, cost of 2**
- **Connected to RouterD on network 18.0.0.0/8, cost of 10**
- **Has a "leaf" network 20.0.0.0/8, cost of 2**

Choosing the Best Path

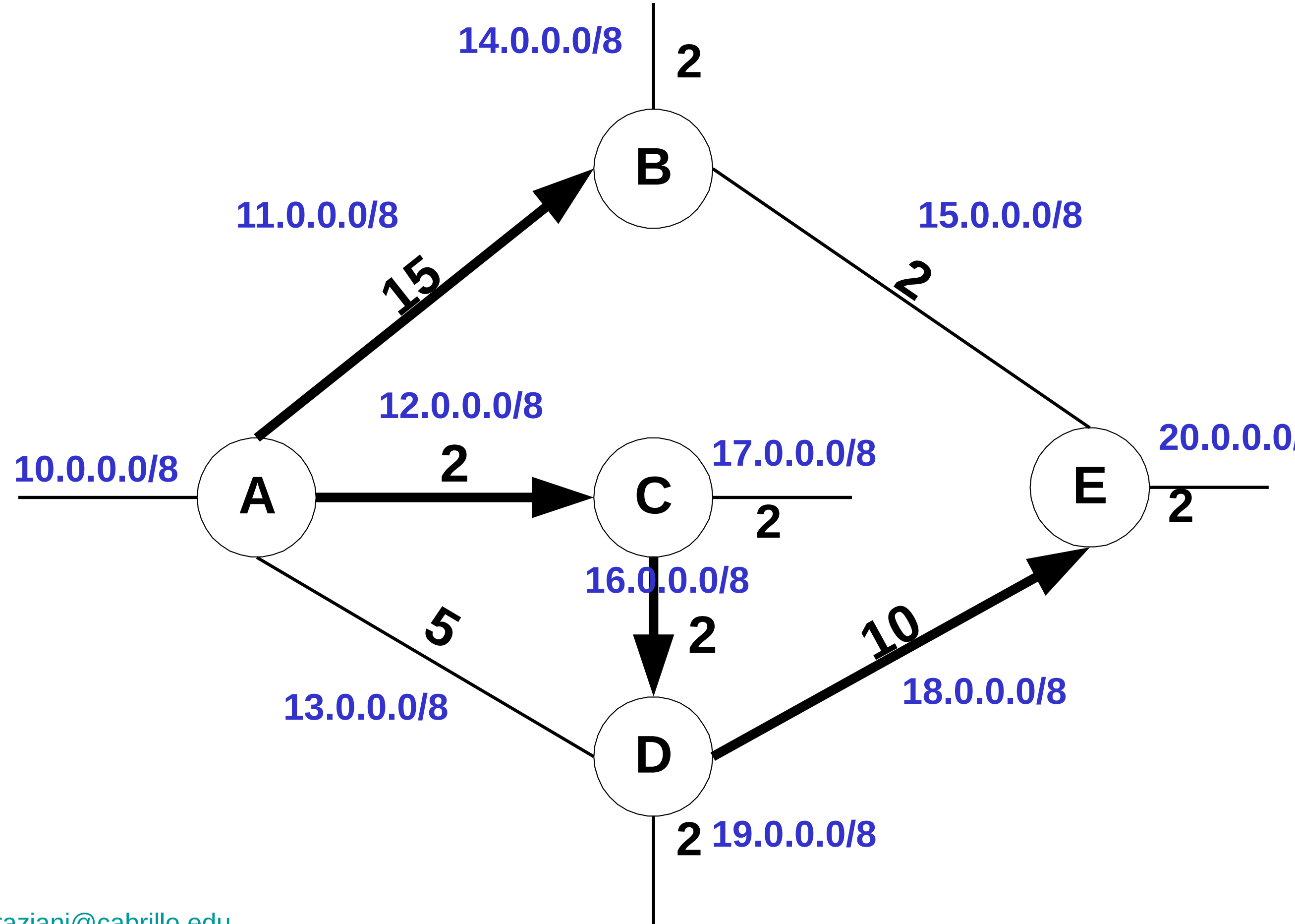
Cabrillo College

- Using the link-state algorithm RouterA can now proceed to find the shortest path to each leaf network.



Choosing the Best Path

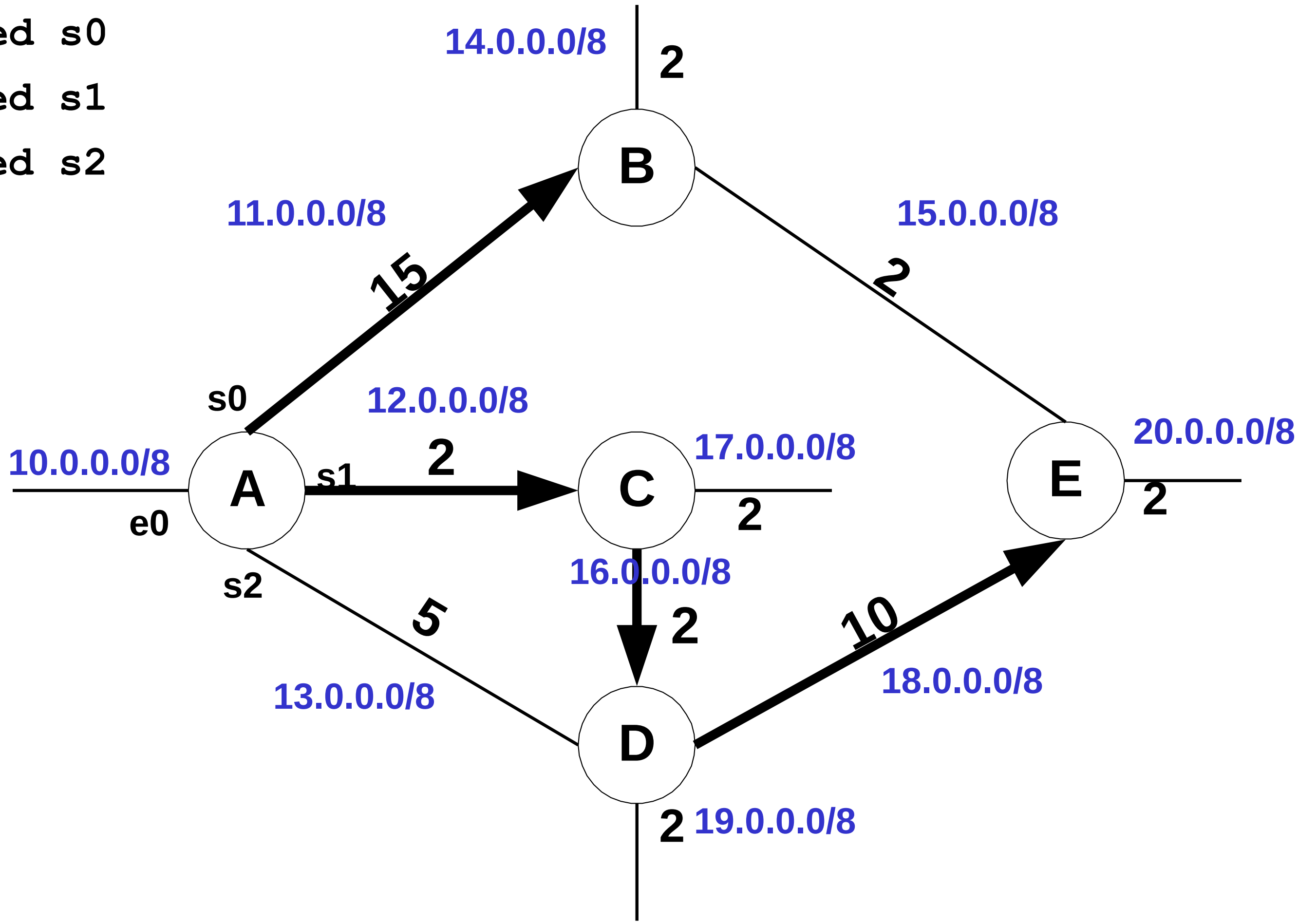
Now RouterA knows the best path to each network, creating an SPT (Shortest Path Tree).



SPT Results Get Put into the Routing Table

RouterA's Routing Table

10.0.0.0/8	connected	e0
11.0.0.0/8	connected	s0
12.0.0.0/8	connected	s1
13.0.0.0/8	connected	s2
14.0.0.0/8	17	s0
15.0.0.0/8	17	s1
16.0.0.0/8	4	s1
17.0.0.0/8	4	s1
18.0.0.0/8	14	s1
19.0.0.0/8	6	s1
20.0.0.0/8	16	s1



Introduction to OSPF Concepts

Introducing OSPF and Link State Concepts

- **Advantages of OSPF**
- **Brief History**
- **Terminology**
- **Link State Concepts**

Introducing the OSPF Routing Protocol

- **Metric based on Cost (Bandwidth)**
- **Hello Protocol**
- **Steps to OSPF Operation**
- **DR/BDR**
- **OSPF Network Types**

Rick Graziani graziani@cabrillo.edu

OSPF's Metric is Cost (Bandwidth)

RFC 2328, OSPF version 2, J. Moy

- **"A cost is associated with the output side of each router interface. This cost is configurable by the system administrator. The lower the cost, the more likely the interface is to be used to forward data traffic."**
- **RFC 2328 does not specify any values for cost.**
- **Bay and some other vendors use a default cost of 1 on all interfaces, essentially making the OSPF cost reflect hop counts.**

OSPF's Metric is Cost (Bandwidth)

Cisco: Cost = Bandwidth

- Cisco uses a default cost of **10^8/bandwidth**
- Default bandwidth of the interface (bandwidth command)
- ***10^8 (100,000,000) as the reference bandwidth***: This is used so that the faster links (higher bandwidth) have lower costs.
  - Routing metrics, lower the cost the better the route.
  - I.e. RIP: 3 hops is better than 10 hops
  - Extra: The reference bandwidth can be modified to accommodate networks with links faster than 100,000,000 bps (100 Mbps). See **`ospf auto-cost reference-bandwidth`** command.
- ***Cost of a route is the cumulative costs of the outgoing interfaces from this router to the network.***

OSPF's Metric is Cost (Bandwidth)

Cisco default interface costs:

- 56-kbps serial link = 1785
- 64-kbps serial link = 1562   128-kbps serial link = 781
- T1 (1.544-Mbps serial link) = 64
- E1 (2.048-Mbps serial link) = 48
- 4-Mbps Token Ring = 25
- Ethernet = 10
- 16-Mbps Token Ring = 6
- Fast Ethernet = 1
- Problem: Gigabit Ethernet and faster = 1

Cost = 100,000,000/Bandwidth

Notes:

- Cisco routers default to T1 (1.544 Mbps) on all serial interfaces and require manual modification with the bandwidth command.
- **`ospf auto-cost reference-bandwidth`** *reference-bandwidth* can be used to modify the reference-bandwidth for higher speed interfaces

OSPF's Metric is Cost (Bandwidth)

Few final notes

- For serial links, if it is not a T1 line, use the bandwidth command to configure the interface to the right bandwidth
- Both sides of the link should have the same bandwidth value
- If you use the command **ospf auto-cost reference-bandwidth *reference-bandwidth***, configure all of the routers to use the same value.

OSPF Packet Types

OSPF Packet Type	Description
Type 1 - Hello	Establishes and maintains adjacency information with neighbors
Type 2 - Database description packet (DBD)	Describes the content of the link-state database on an OSPF router
Type 3 - Link-state request (LSR)	Requests specific pieces of a link-state database
Type 4 - Link-state update (LSU)	Transports link-state advertisements (LSAs) to neighbor routers
Type 5 - Link-state acknowledgement (LSAck)	Acknowledges receipt of a neighbor's LSA

The OSPF Packet Header

Version	Type	Packet Length
Router ID		
Area ID		
Checksum	Authentication Type	
Authentication Data		

OSPF Hello Protocol

Network Mask		
Hello Interval	Options	Router Priority
Dead Interval		
Designated Router		
Backup Designated Router		
Neighbor Router ID		
Neighbor Router ID		
(additional Neighbor Router ID fields can be added to the end of the header, if necessary)		

Hello subprotocol is intended to perform the following tasks within OSPF:

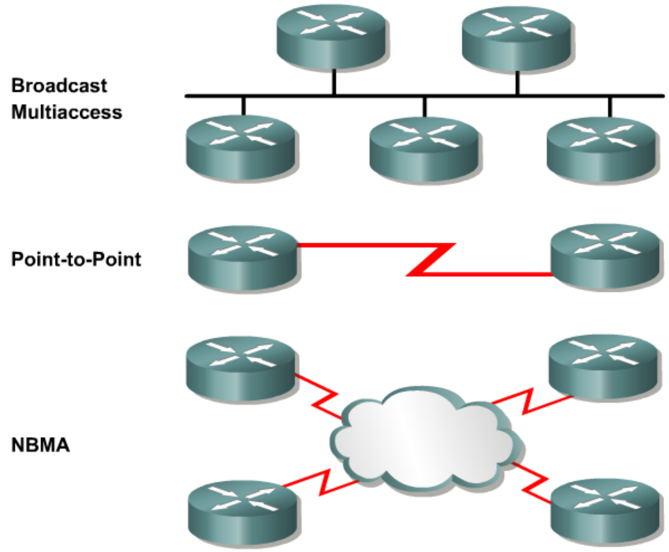
- Dynamic neighbor discovery
- Detect unreachable neighbors
- Ensure two-way communications between neighbors
- Ensure correctness of basic interface parameters between neighbors
- Provide necessary information for the election of the Designated and Backup Designated routers on a LAN segment (coming)

OSPF Hello Protocol

Network Mask		
Hello Interval	Options	Router Priority
Dead Interval		
Designated Router		
Backup Designated Router		
Neighbor Router ID		
Neighbor Router ID		
(additional Neighbor Router ID fields can be added to the end of the header, if necessary)		

- OSPF routers send **Hellos** on OSPF enabled interfaces:
  - –Default every 10 seconds on multi-access and point-to-point segments
  - –Default every 30 seconds on NBMA segments (Frame Relay, X.25, ATM)
  - –Most cases OSPF Hello packets are sent as multicast to ALLSPFRouters (224.0.0.5)
- **HelloInterval** - Cisco default = 10 seconds or 30 seconds and can be changed with the command **`ip ospf hello-interval`**.
- **RouterDeadInterval** - The period in seconds that the router will wait to hear a Hello from a neighbor before declaring the neighbor down.
  - –Cisco uses a default of ***four-times the*** **HelloInterval** (4 x 10 sec. = 40 seconds, 120 secconds for NBMA) and can be changed with the command **`ip ospf dead-interval`**.
- **Note**: For routers to become adjacent, the **Hello**, **DeadInterval** and **network types** must be identical between routers or Hello packets get dropped!

Network Types – *more later*



`show ip ospf interface`

Network Type	Table Title
Broadcast multiaccess	Ethernet, Token Ring, or FDDI
Nonbroadcast multiaccess	Frame Relay, X.25, SMDS
Point-to-point	PPP, HDLC
Point-to-multipoint	Configured by an administrator

Unless you are configuring an NBMA network like Frame Relay, this won't be an issue.

- Many administrators prefer to use **point-to-point** or **point-to-multipoint** for **NMBA** to avoid the DR/BDR and full-mesh issues.

OSPF packet types (Extra)

Type	Description
1	Hello (establishes and maintains adjacency relationships with neighbors)
2	Database description packet (describes the contents of an OSPF router's link-state database) **OSPF Type-2 (DBD)**
3	Link-state request (requests specific pieces of a neighbor router's link-state database) **OSPF Type-3 (LSR)**
4	Link-state update (transports link-state advertisements (LSAs) to neighbor routers) **OSPF Type-4 (LSU)**
5	Link-state acknowledgement (Neighbor routers acknowledge receipt of the LSAs) **OSPF Type-5 (LSAck)**

OSPF packet types (Extra)

Type	Description
1	Hello (establishes and maintains adjacency relationships with neighbors)
2	Database d... router's link-state database)
3	Link-state request (requests ... link-state database)
4	Link-state update (transport... neighbor routers)
5	Link-state acknowledgemen... receipt of the LSAs)

OSPF Type-4 packets have 7 LSA packets

LSA Type	Name	Description
1	Router link entry (record) (O-OSPF)	Generated by each router for each area it belongs to. It describes the states of the router's link to the area. These are only flooded within a particular area.The link status and cost are two of the descriptors provided.
2	Network link entry (O-OSPF)	Generated by Designated Router in multiaccess networks. They describe the set of routers attached to a particular network. LSA Type 2 messages are flooded only within the area that contains the network.
3 or 4	Summary link entry (IA-OSPF Inter area)	Originated by ABRs. They describes the links between the ABR and the internal routers of a local area. These entries are flooded throughout the backbone area to the other ABRs. Type-3 message s describe routes to networks within the local area and are sent to the backbone area. Type-4 messages describe reachability to ASBRs. These link entries are not flooded through totally stubby areas.
5	Autonomous system external link entry (E1-OSPF external type-1)	Originated by the ASBR. Describes routes to destinations external to the autonomous system. Flooded throughout an OSPF autonomous system except for stub and totally stubby areas.

Steps to OSPF Operation

Cabrillo College

Steps of OSPF Operation

- Establish router adjacencies
- Elect a designated router and a backup designated router
- Discover routes
- Select appropriate route to use
- Maintain routing information

OSPF States

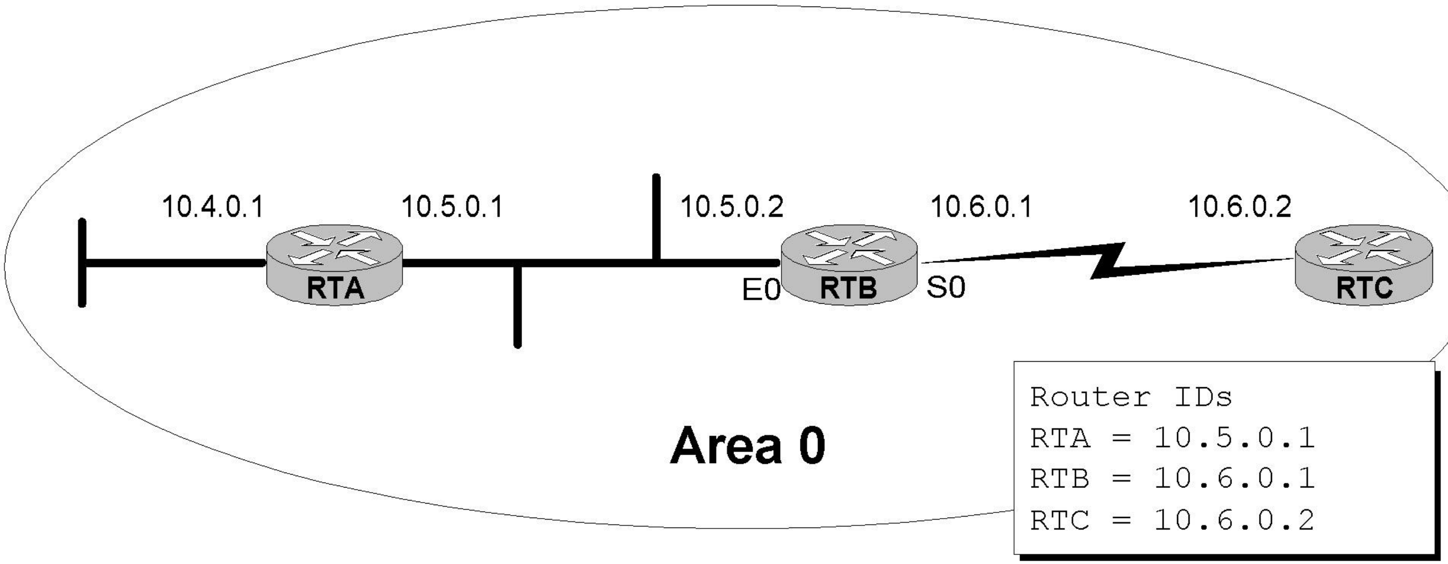
- Down
- Init
- Two-way
- ExStart
- Exchange
- Loading
- Full adjaceny

Steps to OSPF Operation with States

1. **Establishing router adjacencies (Routers are adjacent)**
   - Down State – No Hello received
   - Init State – Hello received, but not with this router's Router ID
     - "Hi, my name is Carlos." "Hi, my name is Maria."
   - Two-way State – Hello received, and with this router's Router ID
     - "Hi, Maria, my name is Carlos." "Hi, Carlos, my name is Maria."
2. **Electing DR and BDR – Multi-access (broadcast) segments only**
   - ExStart State with DR and BDR
   - Two-way State with all other routers
3. **Discovering Routes**
   - ExStart State
   - Exchange State
   - Loading State
   - Full State (Routers are "fully adjacent")
4. **Calculating the Routing Table**
5. **Maintaining the LSDB and Routing Table**

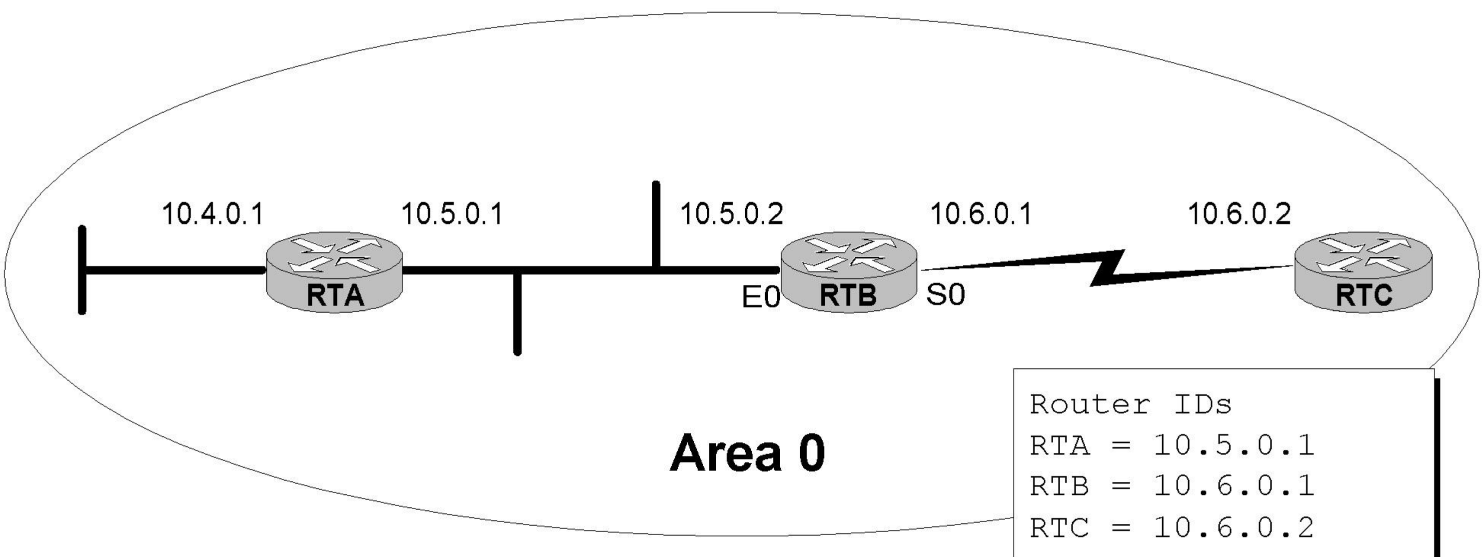
1. Establishing Adjacencies

- Initially, an OSPF router interface is in the **down state**.
- An OSPF interface can transition back to this state if it has not received a Hello packet from a neighbor within the RouterDeadInterval time (40 seconds unless NBMA, 120 seconds).
- In the **down state**, the OSPF process has not exchanged information with any neighbor.
- OSPF is waiting to enter the **init state**.
- An OSPF router tries to form an adjacency with at least one neighbor for each IP network it's connected to.

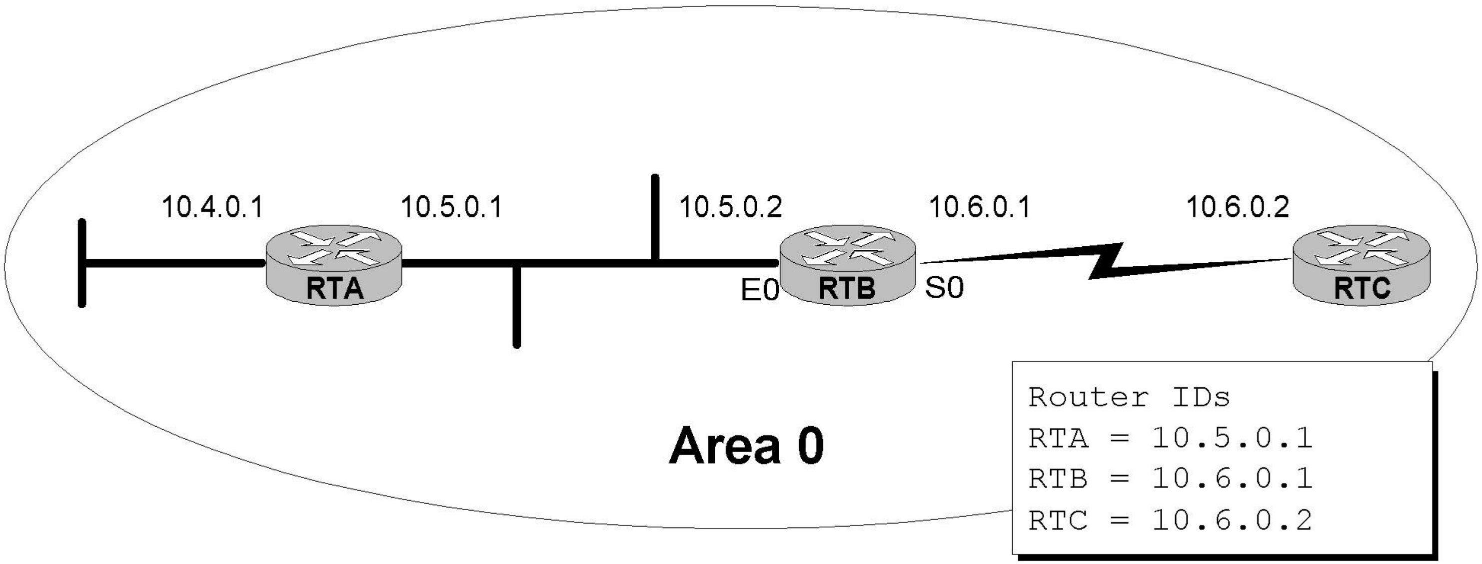


1. Establishing Adjacencies

- The process of establishing adjacencies is asymmetric, meaning the states between two adjacent routers may be different as they both transition to full state.
- RTB perspective and assuming routers are configured correctly.
- Trying to start a relationship and wanting to enter the **init state** or really the **two-way-state**
- RTB begins multicasts **OSPF Hello packets** (**224.0.0.5**, AllSPFRouters), advertising its own **Router ID**.
  - 224.0.0.5: All OSPF routers should be able to transmit and listen to this address.

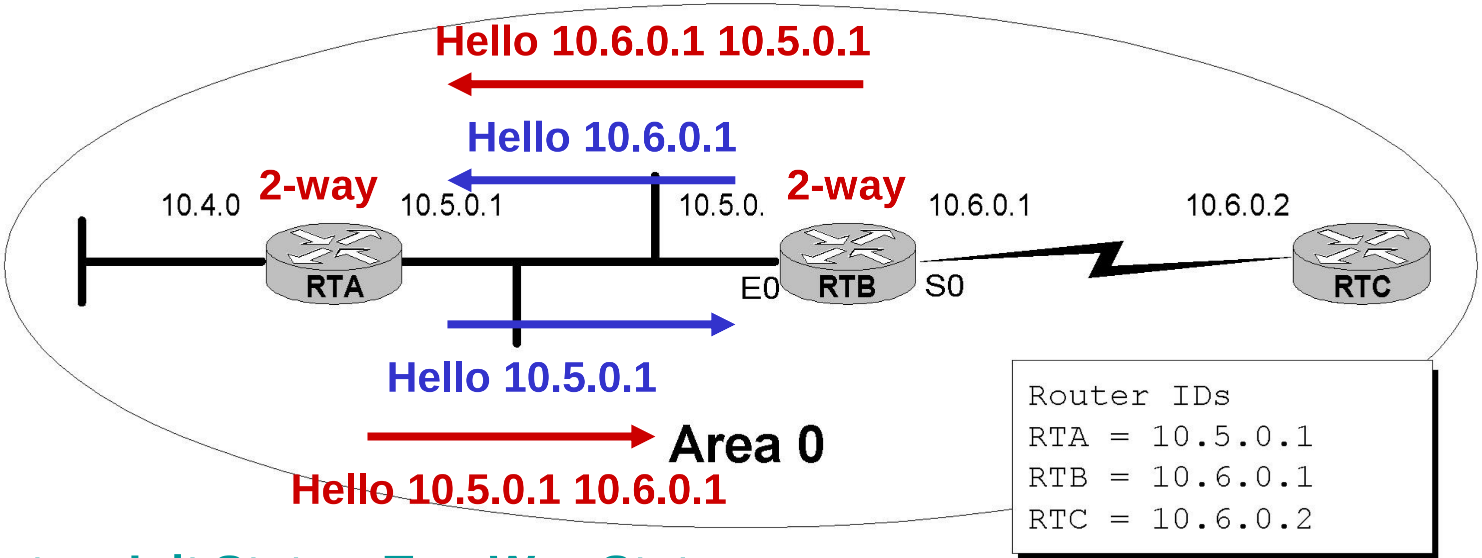


1. Establishing Adjacencies



- **Router ID** = router-id command, else highest loopback address, else highest active IP address.
- **Loopback address** has the advantage of never going down, thus diminishing the possibility of having to re-establish adjacencies. (more in a moment)
  - **Use private ip addresses for loopbacks**, so you do not inadvertently advertise a route to a real network that does not exist on your router.
- **For routers to become adjacent**, the **Hello**, **DeadInterval** and **network types** must be identical between routers or Hello packets get dropped!

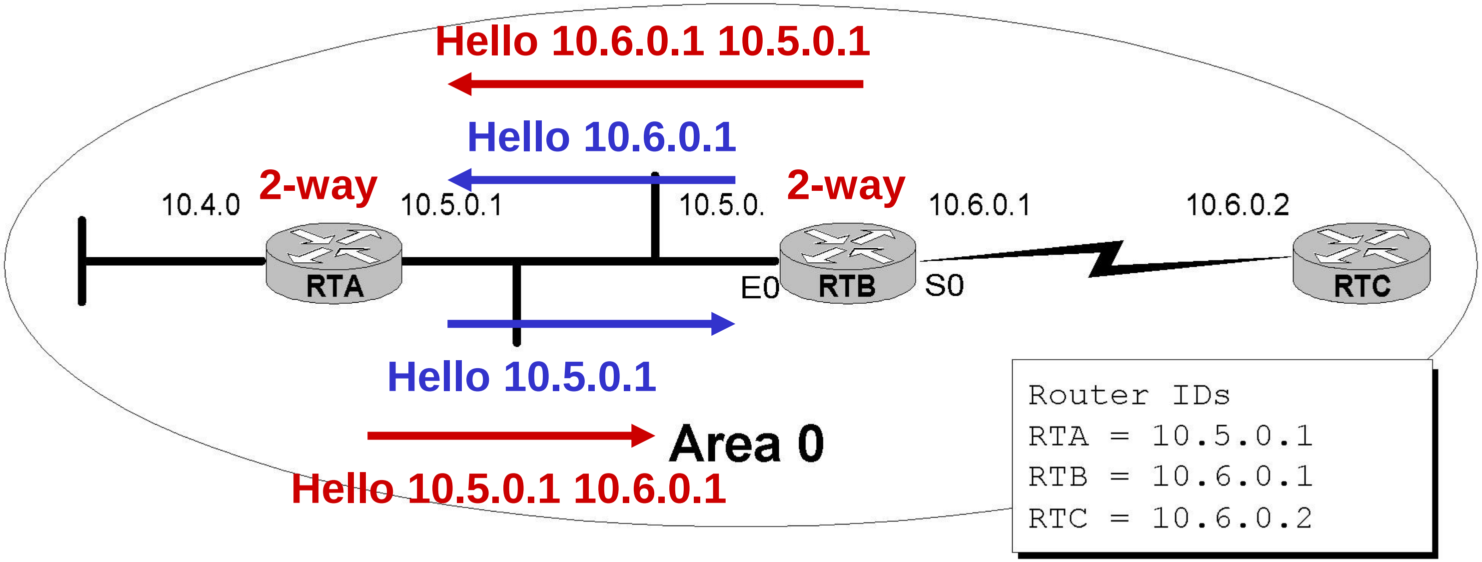
1. Establishing Adjacencies



Down State - Init State – Two Way State

- **Down State** - OSPF routers send Type 1 Hello packets at regular intervals (10 sec.) to establish neighbors.
- When a router (sends or) receives its first **Hello packet**, it enters the **init state**, indicating that the Hello packet was received but did not contain the Router ID of the receiving router in the list of neighbors, so two-way communications is not yet ensured.
- As soon as the router sends a Hello packet to the neighbor with its RouterID and the neighbor sends a Hello packet packet back with that Router ID, the router's interface will transition to the **two-way state**.
- Now, the router is ready to take the relationship to the next level.

1. Establishing Adjacencies



From **Init state** to the **Two-way state**

- RTB receives Hello packets from RTA and RTC (its neighbors), and sees its own Router ID (10.6.0.1) in the Neighbor ID field.
- RTB declares takes the relationship to a new level, and declares a **two-way state** between itself and RTA, and itself and RTC.
- As soon as the router sends a **Hello packet** to the neighbor with its RouterID and the neighbor sends a Hello packet packet back with that Router ID, the router's interface will transition to the two-way state.
- Now, the router is ready to take the relationship to the next level.

1. Establishing Adjacencies

Two-way state

- RTB now decides who to establish a "full adjacency" with depending upon the type of network that the particular interfaces resides on.
- **Note**: The term adjacency is used to both describe routers reaching 2-way state and when they reach full-state. Not to go overboard on this, but technically OSPF routers are adjacent when the FSM reaches full-state and IS-IS is considered adjacent when the FSM reaches 2-way state.

Two-way state to ExStart state

- If the interface is on a point-to-point link, the routers becomes adjacent with its sole link partner (aka "soul mates"), and take the relationship to the next level by entering the **ExStart state**. (coming soon)

Remaining in the two-way state

- If the interface is on a multi-access link (Ethernet, Frame Relay, …) RTB must enter an election process to see who it will establish a full adjacency with, and remains in the **two-way state**. (Next!)

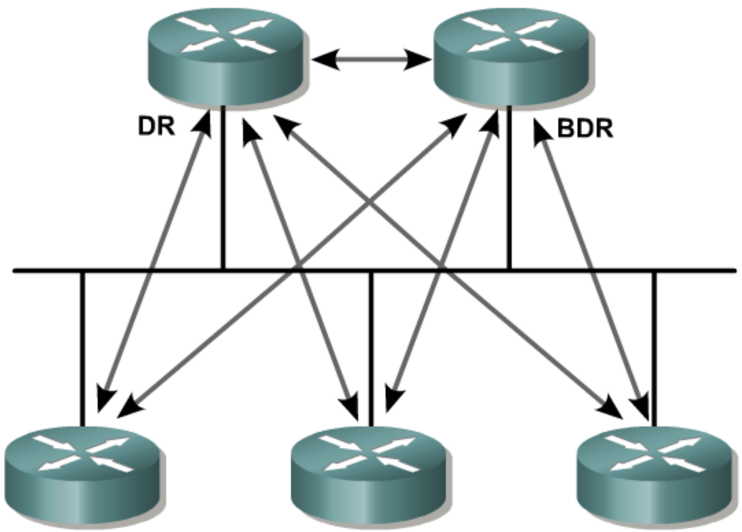
Steps to OSPF Operation with States

1. **Establishing router adjacencies (Routers are adjacent)**
   - Down State – No Hello received
   - Init State – Hello received, but not with this router's Router ID
     - "Hi, my name is Carlos." "Hi, my name is Maria."
   - Two-way State – Hello received, and with this router's Router ID
     - "Hi, Maria, my name is Carlos." "Hi, Carlos, my name is Maria."
2. **Electing DR and BDR – Multi-access (broadcast) segments only**
   - ExStart State with DR and BDR
   - Two-way State with all other routers
3. **Discovering Routes**
   - ExStart State
   - Exchange State
   - Loading State
   - Full State (Routers are "fully adjacent")
4. **Calculating the Routing Table**
5. **Maintaining the LSDB and Routing Table**

Electing the DR and BDR

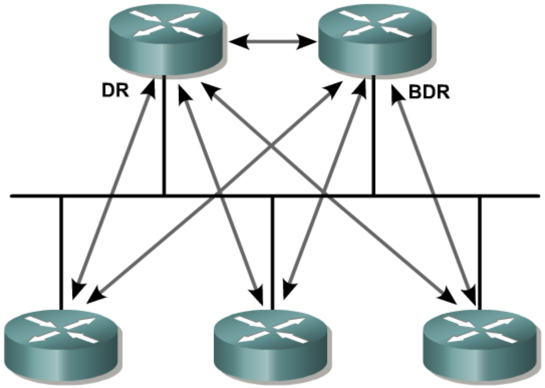
Cabrillo College

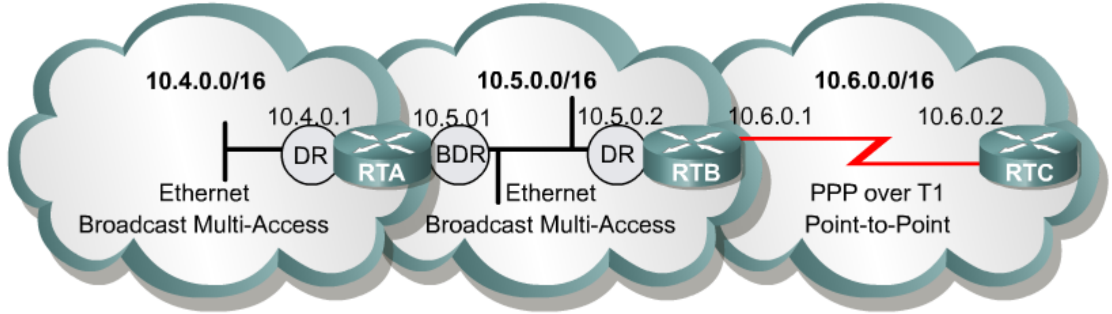
- On multi-access, broadcast links (Ethernet), a DR and BDR (if there is more than one router) need to be elected.
- ***DR*** - Designated Router
- ***BDR*** – Backup Designated Router
- DR's serve as collection points for Link State Advertisements (LSAs) on multi-access networks
- A BDR back ups the DR.
- If the IP network is ***multi-access***, the OSPF routers will elect one DR and one BDR



- **Without a DR, the formation of an adjacency between every attached router would create many unnecessary LSA (Link State Advertisements), n(n-1)/2 adjacencies.**
- **Flooding on the network itself would be chaotic.**

Electing the DR and BDR





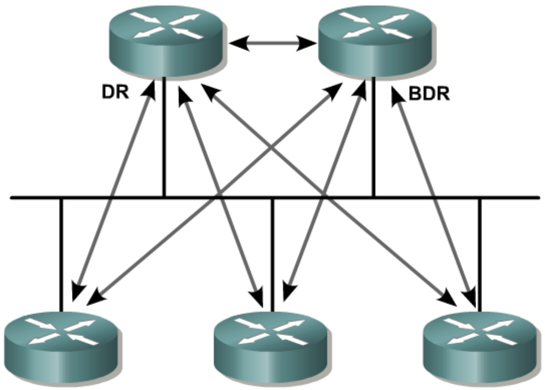
- Router with the **highest Router ID** is elected the DR, next is BDR.
- But like other elections, this one can be rigged.
- The router's priority field can be set to either ensure that it becomes the DR or prevent it from being the DR.

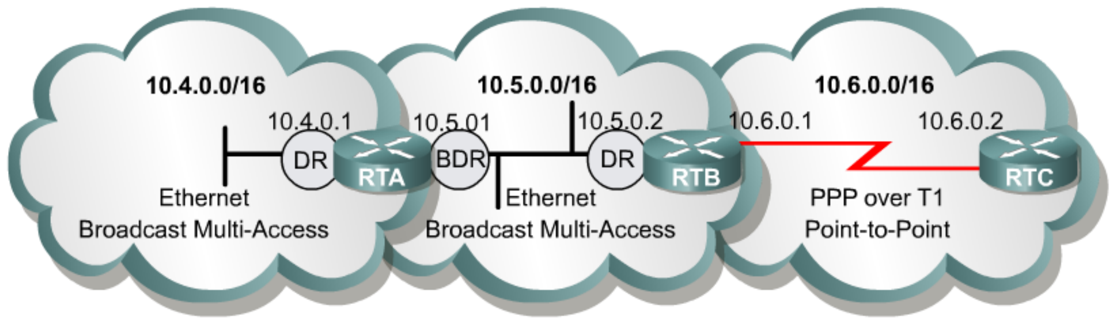
```
Rtr(config-if)# ip ospf priority <0-255>
```

  – Higher priority becomes DR/BDR
  – Default = 1
  – 0 = Ineligible to become DR/BDR
- The router can be assigned a priority between 0 and 255, with 0 preventing this router from becoming the DR (or BDR) and 255 ensuring at least a tie. (The highest Router ID would break the tie.)

Electing the DR and BDR

Cabrillo College



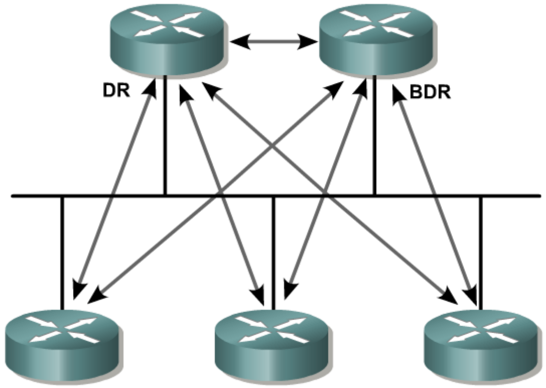


- All other routers, "**DROther**", establish adjacencies with only the DR and BDR.
- DRother routers multicast LSAs to only the DR and BDR
  - (224.0.0.6 - all DR routers)
- DR sends LSA to all adjacent neighbors (DROthers)
  - (224.0.0.5 - all OSPF routers)

Backup Designated Router - BDR

- Listens, but doesn't act.
- If LSA is sent, BDR sets a timer.
- If timer expires before it sees the reply from the DR, it becomes the DR and takes over the update process.
- The process for a new BDR begins.

Electing the DR and BDR



A new router enters the network:

- Once a DR is established, a new router that enters the network with a higher priority or Router ID it will **NOT** become the DR or BDR. (Bug in early IOS 12.0)
- Regardless of the priority or Router ID, that router will become a DROther.
- If DR fails, BDR takes over as DR and selection process for new BDR begins.

Clarifications

- **Hello packets** are still exchanged between all routers on a multi-access segment (DR, BDR, DROthers,….) to maintain neighbor adjacencies.
- **OSPF LSA packets** (coming) are packets which are sent from the BDR/DROthers to the DR, and then from the DR to the BDR/DROthers. (The reason for a DR/BDR.)
- Normal routing of **IP packets** still takes the lowest cost route, which might be between two DROthers.

Steps to OSPF Operation with States - Extra

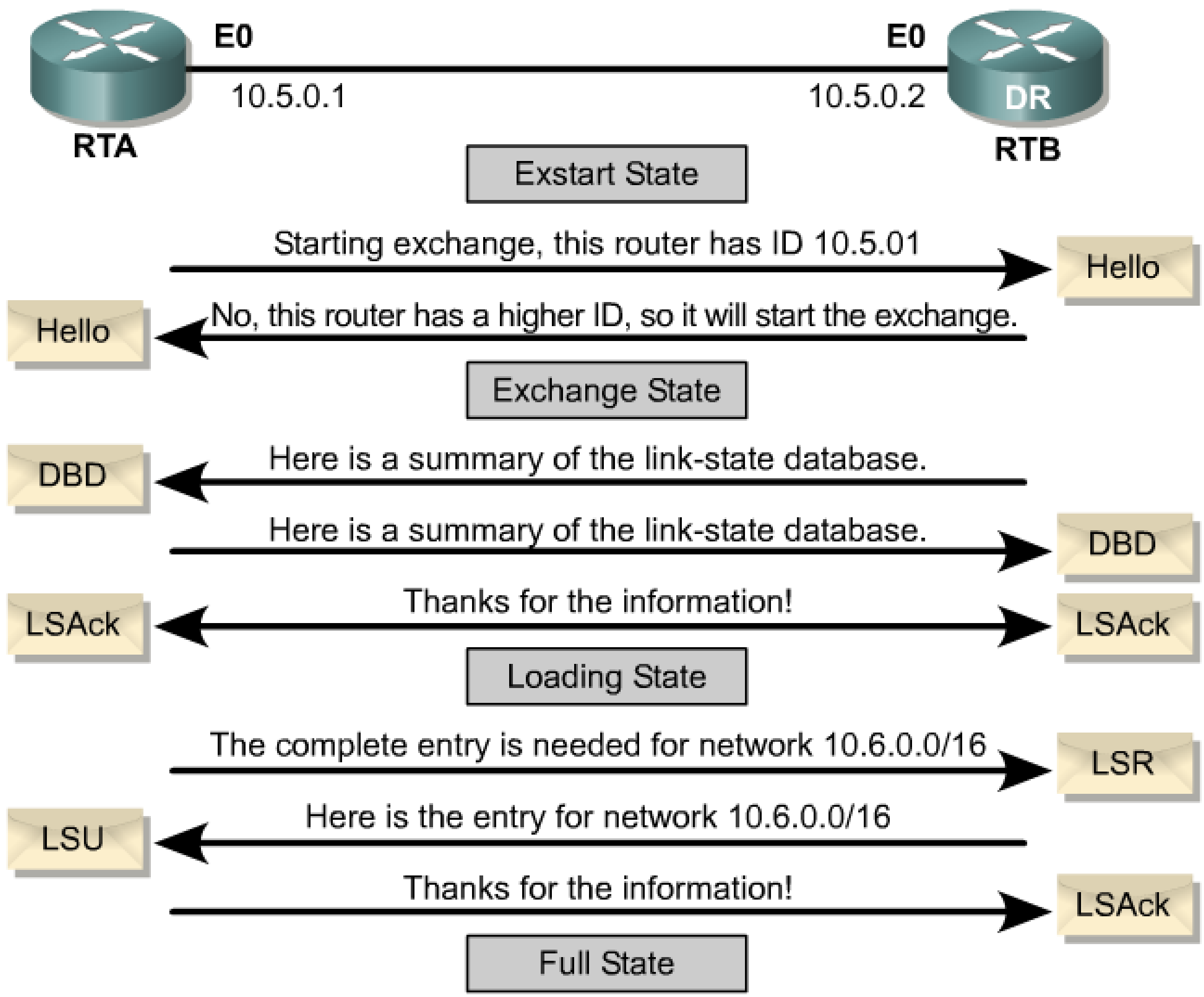
1. Establishing router adjacencies
   - Down State – No Hello received
   - Init State – Hello received, but not with this router's Router ID
     - "Hi, my name is Carlos." "Hi, my name is Maria."
   - Two-way State – Hello received, and with this router's Router ID
     - "Hi, Maria, my name is Carlos." "Hi, Carlos, my name is Maria."
2. Electing DR and BDR – Multi-access (broadcast) segments only
   - ExStart State with DR and BDR
   - Two-way State with all other routers
3. Discovering Routes
   - ExStart State
   - Exchange State
   - Loading State
   - Full State

4. Calculating the Routing Table

5. Maintaining the LSDB and Routing Table

Steps to OSPF Operation with States
Discovering Routes and Reaching Full State

Cabrillo College



ExStart State – the explanation

ExStart State

- This state starts the LSDB (Link State Data Base) synchronization process.
- This will prepare for initial database exchange.
- Routers are now ready to exchange routing information.
  - Between routers on a point-to-point network
  - On a multi-access network between the DRothers and the DR and BDR.
- Formally, routers in **ExStart state** are characterized as adjacent, but have not yet become “fully adjacent” as they have not exchanged data base information.

But who goes first in the exchange?

- **ExStart** is established by exchanging OSPF Type-2 DBD (Database Description) packets (I believe the curriculum says LSA type 2 which is something else).
- Purpose of **ExStart** is to establish a “***master/slave relationship”*** between the two routers decided by the **higher router id**.
- Once the roles are established they enter the **Exchange state**.

DBD Packet

OSPF Packet Type	Description
Type 1 - Hello	Establishes and maintains adjacency information with neighbors
Type 2 - Database description packet (DBD)	Describes the content of the link-state database on an OSPF router
Type 3 - Link-state request (LSR)	Requests specific pieces of a link-state database
Type 4 - Link-state update (LSU)	Transports link-state advertisements (LSAs) to neighbor routers
Type 5 - Link-state acknowledgement (LSAck)	Acknowledges receipt of a neighbor's LSA

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Version #   |       2       |         Packet length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Router ID                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Area ID                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Checksum            |             AuthType            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Authentication                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Authentication                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Interface MTU         |    Options    |0|0|0|0|R|I|M|MS
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     DD sequence number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-                                                             -+
|                                                               |
+-                      An LSA Header                          -+
|                                                               |
+-                                                             -+
|                                                               |
+-                                                             -+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              ...                              |
```

(LSA descriptions)

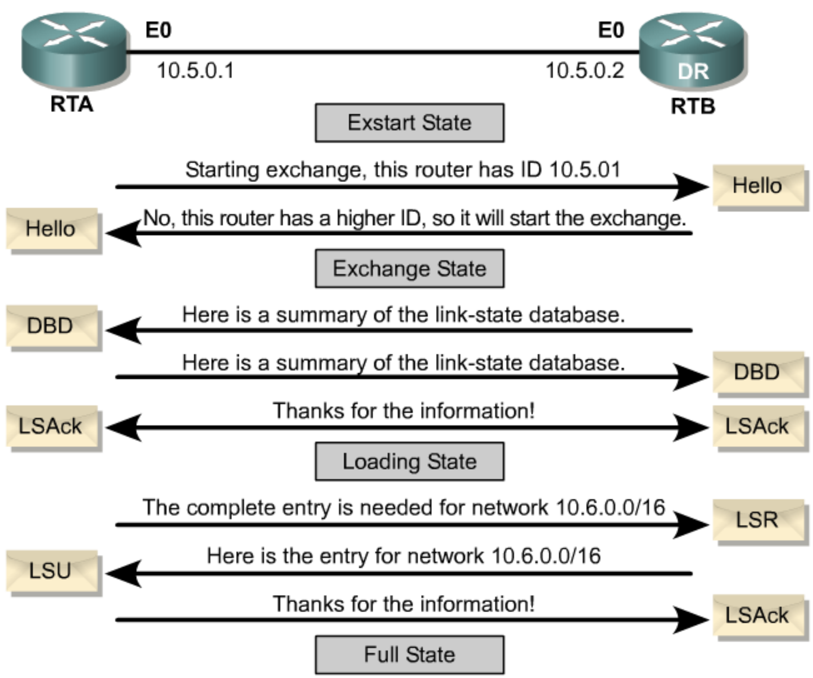
Exchange State – the explanation

- Routers exchange one or more Type-2 DBDs (Database Description) packets, which is a **summary of the link-state database**
  - send LSAcks to verify
- Routers compare these DBDs with information in its own database.
- When a DBD packet is received the router looks through the LSA (Link State Advertisement) headers and identifies LSAs that are not in the router's LSDB or are a different version from its LSDB version (older or newer).
- If the LSA is not in its LSDB or the LSA is a more recent version, the router adds an entry to its **Link State Request list**.
- This process ends when both routers stop have sent and received acknowledgements for all their DBD packets – that is they have successfully sent all their DBD packets to each other.

Exchange State – the explanation

Exchange State

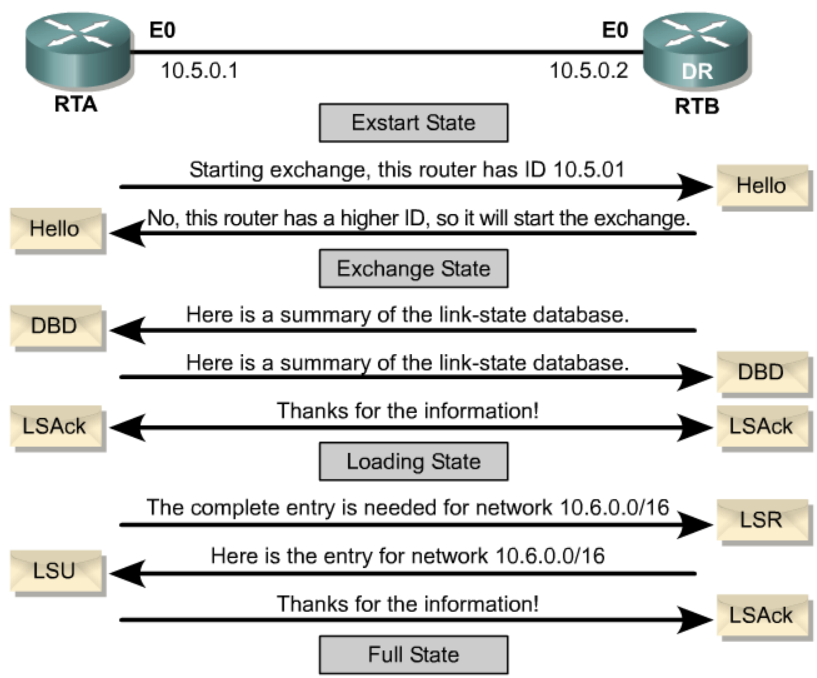
- If a router has entries in its **Link State Request list**, meaning that it needs additional information from the other router for routes that are not in its LSDB or has more recent versions, then it enters the **loading state.**
- If there are no entries in its **Link State Request list**, than the router's interface can transition directly to **full state**.
- Complete routing information is exchanged in the **loading state**, discussed next.



Loading State - the explanation

Loading State

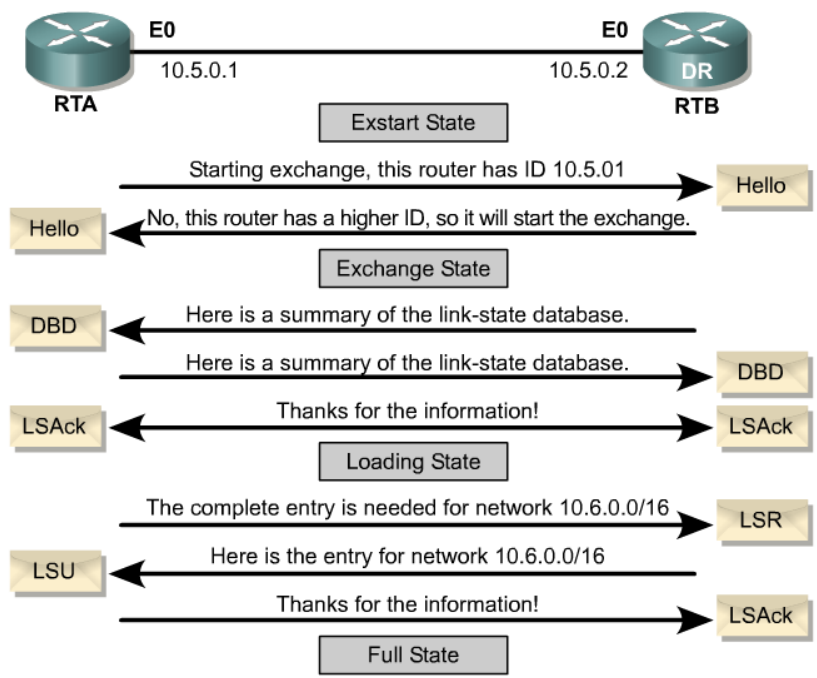
- If a router has entries in its **Link State Request list**, meaning that it needs additional information from the other router for routes that are not in its LSDB or has more recent versions, then it enters the **loading state.**
- The router needing additional information sends **LSR (Link State Request) packets** using LSA information from its LSR list.



Loading State - the explanation

Loading State

- The other routers replies by sending the requested LSAs in the **Link State Update (LSU) packet.**
- The receiving router sends **LSAck** to acknowledge receipt.
- When all LSAs on the neighbors **Link State Request list** have been received, the “neighbor FSM” transitions this interface to **Full state.**



Link State Requests (LSR)

Cabrillo College

```
 0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   Version #   |       3       |         Packet length         |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                          Router ID                            |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                           Area ID                             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |           Checksum            |             AuType            |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                       Authentication                          |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                       Authentication                          |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                          LS type                              |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                       Link State ID                           |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                     Advertising Router                        |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                              ...                              |
```

LSR

(LSRs)

Link State Advertisement (Update)

Cabrillo College

```
 0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   Version #   |       4       |         Packet length         |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                          Router ID                            |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                           Area ID                             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |           Checksum            |             AuType            |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                       Authentication                          |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                       Authentication                          |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                            # LSAs                             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
 +-                                                            +-+
 |                             LSAs                              |
 +-                                                            +-+
 |                              ...                              |
```

LSAs: Types 1, 2, 3, 4, or 5

LSAs

Full State - the explanation

Full State

- **Full state** - after all LSRs have been updated.
- At this point the routers should have identical LSDBs (link-state databases).

Flooding LSAs

- Once this interface transitions to or from **Full state** the router **originates** a new version of a **Router LSA** (coming) and **floods it to its neighbors**, distributing the new topological information – out all OSPF enabled interfaces.
- Broadcast networks:
  - DR: If the LSA was received on this interface, send it out this interface so DROthers receive it (224.0.0.5 - all OSPF routers)
  - BDR/DROther: If the LSA was received on this interface, do <u>not</u> send out this interface (received from DR).

Calculating Routing Table

- The router still must calculate its routing table – ***Next!***

Couple of notes on link state flooding…

- OSPF is a link state routing protocol and **does <u>not</u> send periodic updates** like RIP.
- OSPF only **floods** link state state advertisements when there is a **change in topology** (this includes when a routers are first booted).
- OSPF uses **hop-by-hop flooding** of LSAs; an LSA received on one interface are flooded out other OSPF enabled interfaces.
- If a link state entry in the LSDB (Link State DataBase) reaches an age of **60 minutes (MaxAge)** without being updated, it is removed and SPF is recalculated.
- Every **30 minutes (LSRefreshTime),** OSPF routers flood only their link states to all other routers (in the area).
  - This is known as a **"paranoid update"**
  - These do not trigger SPF recalculations.
- Special note: When a **link goes down** and a router wants to send a LSA to tell other routers to remove this link state, it sends this link state with a **value of 60 minutes (MAXAGE).**

Steps to OSPF Operation with States - Extra

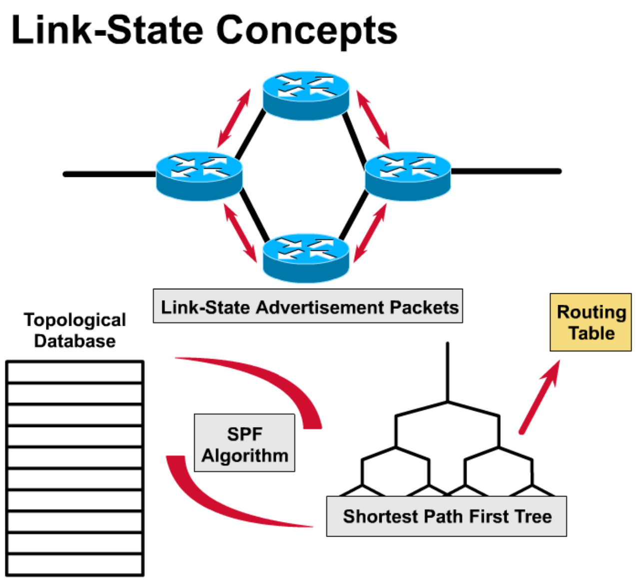
1. Establishing router adjacencies
   - Down State – No Hello received
   - Init State – Hello received, but not with this router's Router ID
     - "Hi, my name is Carlos." "Hi, my name is Maria."
   - Two-way State – Hello received, and with this router's Router ID
     - "Hi, Maria, my name is Carlos." "Hi, Carlos, my name is Maria."
2. Electing DR and BDR – Multi-access (broadcast) segments only
   - ExStart State with DR and BDR
   - Two-way State with all other routers
3. Discovering Routes
   - ExStart State
   - Exchange State
   - Loading State
   - Full State

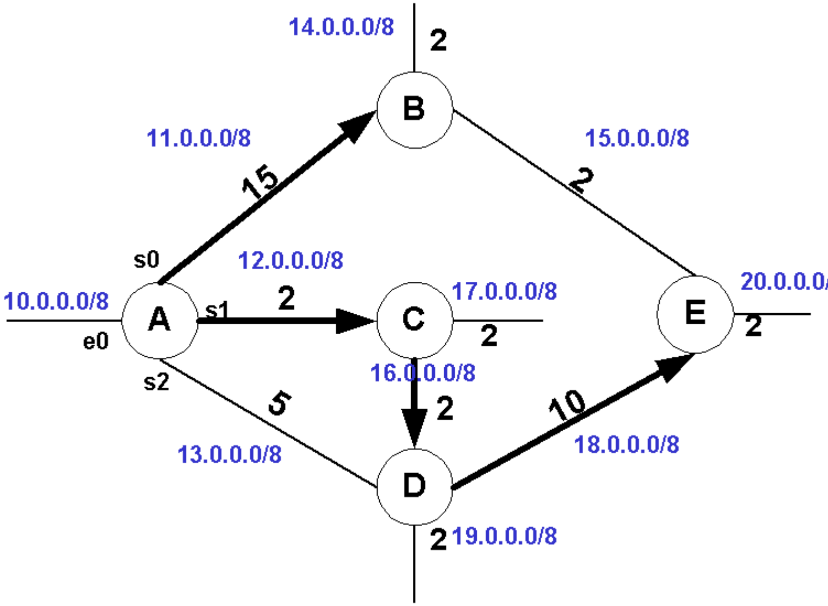
4. Calculating the Routing Table

5. Maintaining the LSDB and Routing Table

4. Calculating the Routing Table

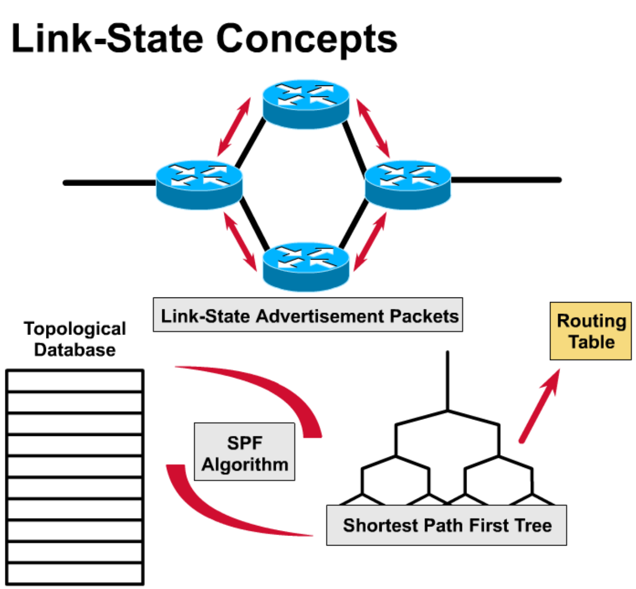
- The router now has a complete link-state database
- Now the router is ready to create a routing table, but first needs to run the Shortest Path First Algorithm on the link state database, which will create the SPF tree.
- **Dijkstra's algorithm** is used to calculate the Shortest Path Tree from the LSAs in the link state database.
- SPF, Shortest Path First calculations places itself as the root and creating a "tree diagram" of the network.

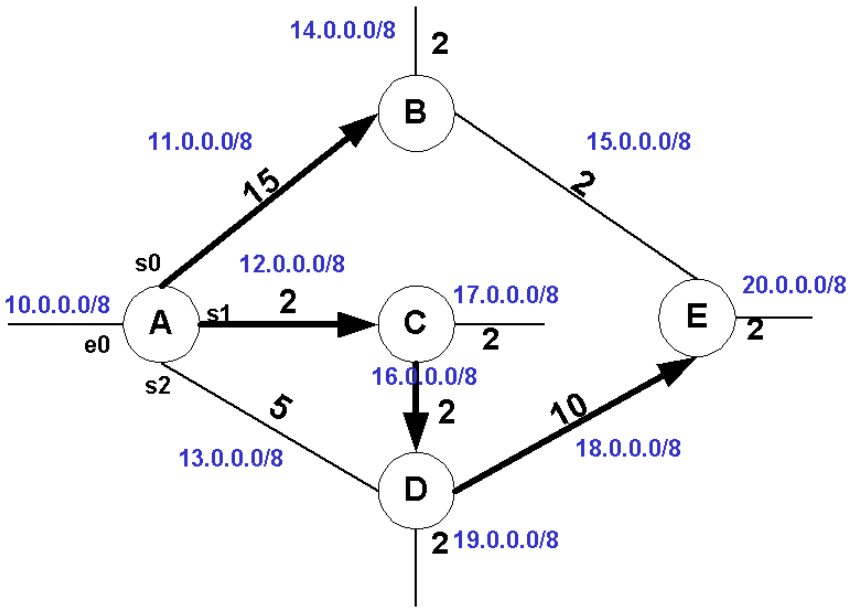




4. Calculating the Routing Table

- The LSAs that build the database contain three important pieces of generic information: **RouterID** of the sender of the LSA, the **NeighborID**, and **cost** of the link between the Router and the neighbor (I.e the state of the link or link-state).
- We will not go into the details here, but the books mentioned earlier all some excellent examples on this process.
- Also, remember the link-state exercise we did earlier!





4. Calculating the Routing Table

SPF Holdtime

- SPF algorithm is CPU intensive and takes some time depending upon the size of the area (coming next week), the number of routers, the size of the link state database.
- A flapping link can cause an OSPF router to keep on recomputing a new routing table, and never converge.
- To minimize this problem:
  - SPF calculations are **delayed by 5 seconds** after **receiving an LSU** (Link State Update)
  - **Delay between consecutive SPF calculations** is **10 seconds**
- You can configure the delay time between when OSPF receives a topology change and when it starts a shortest path first (SPF) calculation (*spf-delay*).
- You can also configure the hold time between two consecutive SPF calculations (*spf-holdtime*).

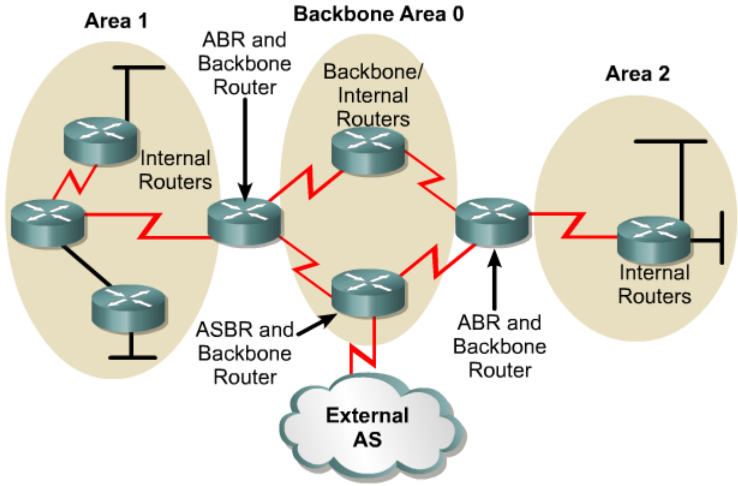
```
Router(config-router)#timers spf spf-delay spf-holdtime
```

4. Calculating the Routing Table

Cabrillo College

```
RTB#show ip ospf 1
Routing Process "ospf 1" with ID 10.6.0.1
<OUTPUT OMITTED>
    Area BACKBONE(0)
        Number of interfaces in this area is 2
        Area has no authentication
        SPF algorithm executed 5 times
        Area ranges are
        Number of LSA 4. Checksum Sum 0x1D81A
        Number of opaque link LSA 0. Checksum Sum 0x0
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

4. Calculating the Routing Table



- In the next chapter we will discuss OSPF and multiple areas.
- Here is some information regarding the routing table calculation that we will discuss again in the chapter on OSPF multiple areas:
- OSPF areas are designed to keep issues like flapping links within an area.
- SPF is <u>not</u> recalculated if the topology change is in another area.
- The interesting thing is that OSPF distributes inter-area (between areas) topology information using a distance-vector method.
- **OSPF uses link-state principles only within an area**.
- **ABRs relay routing information between areas via distance vector technique similar to RIP or IGRP**.

4. Calculating the Routing Table

FYI: The rest of the story, which will be discussed in OSPF multiple areas.

OSPF areas are designed to keep issues like flapping links within an area. SPF is not recalculated if the topology change is in another area. The interesting thing is that OSPF distributes inter-area (between areas) topology information using a distance-vector method. OSPF uses link-state principles only within an area. ABRs do not announce topological information between areas, instead, only routing information is injected into other areas. ABRs relay routing information between areas via distance vector technique similar to RIP or IGRP. This is why show ip ospf does not show a change in the number of times SPF has been executed when the topology change is in another area.

Note: It is still a good idea to perform route summarization between areas, announcing multiple routes as a single inter-area route. This will hide any changes in one area from affecting routing tables in other areas.

For more information, look at Cisco IP Routing by Alex Zinin.

4. Calculating the Routing Table - FYI

- **Note**: There is a routing table which is internal to the OSPF process. This internal routing table contains information used as an intermediate result for inter-area and external route calculations and contains routes to ABRs and ASBRs. (Just a technical note and fyi.)

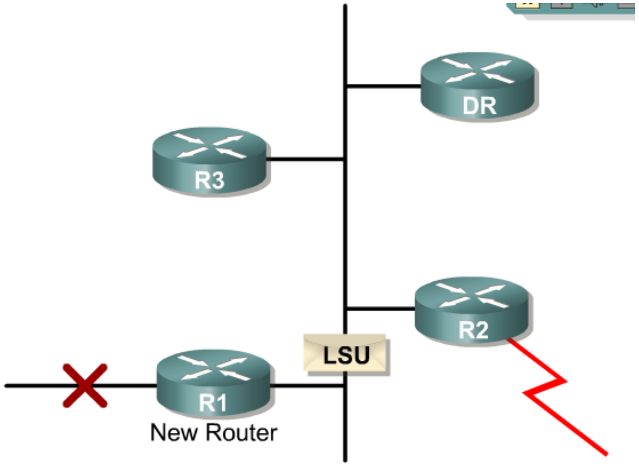
Steps to OSPF Operation with States - Extra

1. Establishing router adjacencies
   - –Down State – No Hello received
   - –Init State – Hello received, but not with this router's Router ID
     - •"Hi, my name is Carlos."   "Hi, my name is Maria."
   - –Two-way State – Hello received, and with this router's Router ID
     - •"Hi, Maria, my name is Carlos."   "Hi, Carlos, my name is Maria."
2. Electing DR and BDR – Multi-access (broadcast) segments only
   - –ExStart State with DR and BDR
   - –Two-way State with all other routers
3. Discovering Routes
   - –ExStart State
   - –Exchange State
   - –Loading State
   - –Full State

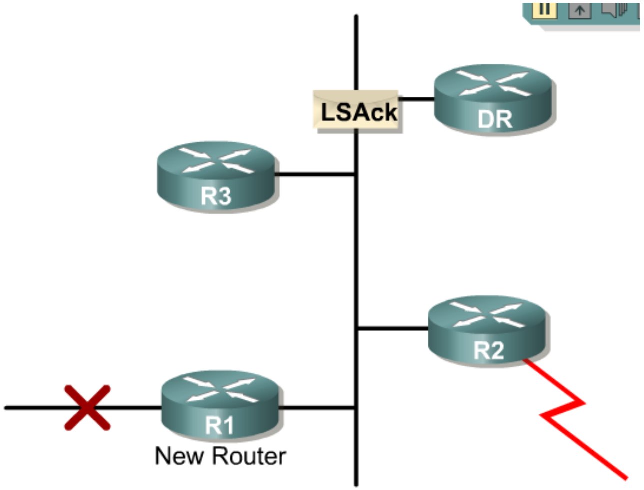
4. Calculating the Routing Table

5. Maintaining the LSDB and Routing Table

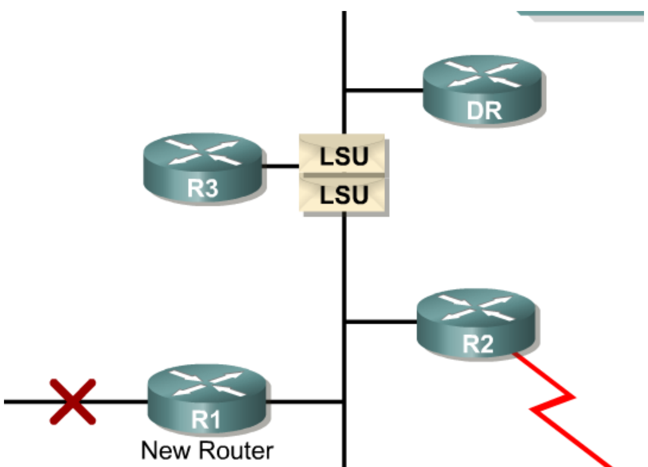
Step 5 – Maintaining LSDB and Routing Table Information



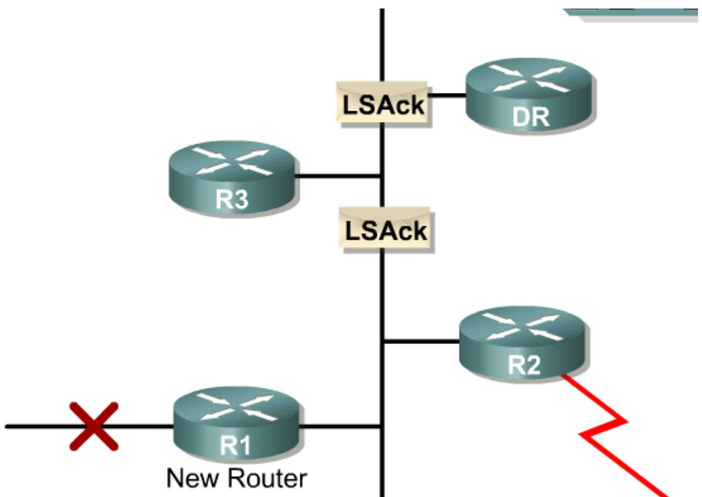
R1 sends out LSU to DR



DR sends ACK to R1

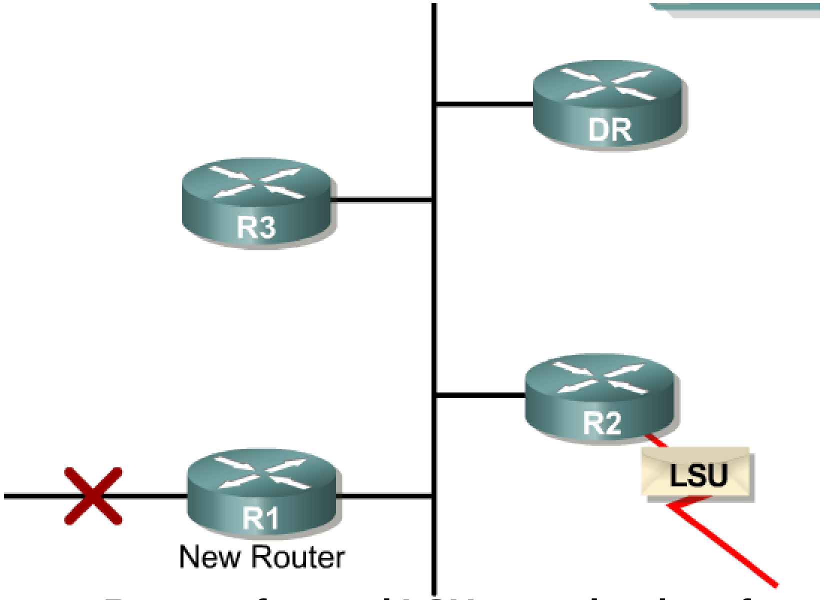


DR sends out LSU to DROthers
(Note graphic should include R1)

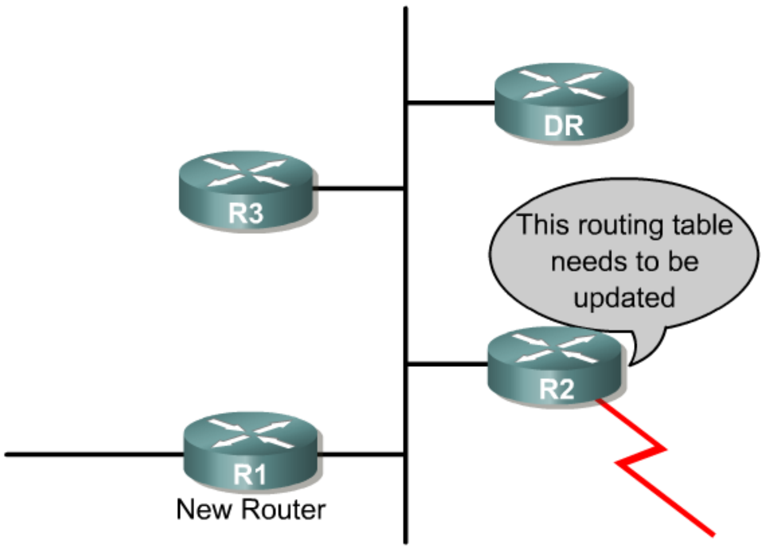


DRothers reply with ACK to DR
(Note graphic should include R1)

Step 5 – Maintaining LSDB and Routing Table Information



Routers forward LSU out other interfaces



Routers rerun SPF to calculate a new routing table

Step 5 – Maintaining LSDB and Routing Table Information

OSPF convergence time for **intra-area** routing is determined by the amount of time routers spend on:

- Link-failure or neighbor unreachability detection
- Origination of the new LSA
- Flooding the new version of the LSA to all routers
- SPF calculation on all routers

When **inter-area** routing is considered, installation or removal of a route in the routing table may trigger the need to send LSAs to other areas.

- New inter-area routes may need to be calculated in the other areas.
- Remember, OSPF distributes inter-area (between areas) topology information using a distance-vector method.
- OSPF uses link-state principles only within an area, so changes in other areas to <u>not</u> cause the router to re-run the SPF algorithm.

Link-failure or neighbor unreachability detection

- In OSPF, link failure can be determined by:
  - Physical layer or data link layer – directly reporting a state change on a directly connected interface.
  - The Hello subprotocol – The router's interface has not received a Hello packet from an adjacent neighbor within the OSPF **RouterDeadInterval** time (40 seconds or 120 seconds on NBMA links).

Step 5 – Maintaining LSDB and Routing Table Information

- FYI: LSAs are not originated any faster than every 5 seconds (MinLSInterval) to prevent flooding storms in unstable networks.
- When the router wants to report a down link, it sets the LS Age field to the MaxAge value (3,600 seconds), which tells routers to flush this entry from their LSDB.

LSU packet

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Version #   |       4       |         Packet length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Router ID                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Area ID                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Checksum            |             AuType            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Authentication                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Authentication                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            # LSAs                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-                                                            +-+
|                             LSAs                              |
+-                                                            +-+
|                              ...                              |
```

Router LSA

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            LS age             |     Options   |       1       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Link State ID                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Advertising Router                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     LS sequence number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         LS checksum           |             length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    0    |V|E|B|        0      |            # links            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Link ID                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Link Data                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     # TOS     |            metric             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              ...                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      TOS      |        0      |          TOS  metric          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Link ID                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Link Data                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Periodic updates

- Each LSA entry in the link-state database has its own age timer, with a default of **60 minutes** (3,600 seconds). – this is known as the **MaxAge** value of the LSA entry.
- When an LSA reaches MaxAge, it is flushed from the LSDB.
- Before this happens the LSA has a **Link State Refresh Time (LSRefreshTimer)**, **30 minutes**, (1,800 seconds) and when this time expires **the router that originated the LSA** will floods a new LSA to all its neighbors, who will reset the age of the LSA in its LSDB.
- This is also known as the "**paranoid update**." or "**periodic update**."
- These updates do not trigger recalculation of the routing table.

Configuring Single Area OSPF

It’s easy!

Cabrillo College

Enabling OSPF

Rtr(config)# `router ospf` *`process-id`*

- **process-id: 1 - 65,535**
- Cisco feature, which allows you to run multiple, different OSPF routing processes on the same router. *(But don't!)*
- **Process-id is locally significant**, and does not have to be the same number on other routers (they don't care).
- This is different than the process-id used for IGRP and EIGRP which must be the same on all routers sharing routing information.
- **Extra**: FYI - Cisco IOS limits the number of dynamic routing processes to 30. This is because it limits the number of protocol descriptors to 32, using one for connected route sources, one for static route sources, and 30 for dynamic route sources.

Configuring the Network Command

Rtr(config)# **router ospf *process-id***

Rtr(config-router)#**network *address wildcard-mask* area *area-id***

- Tells OSPF which interfaces to enable OSPF on (send and receive updates), matching the address and wildcard mask.
- Also, tells OSPF to include this network in its routing updates
- Wildcard is necessary because OSPF supports CIDR and VLSM
- Most of the time you can just use an inverse-mask (like access-lists) as the network wildcard mask.

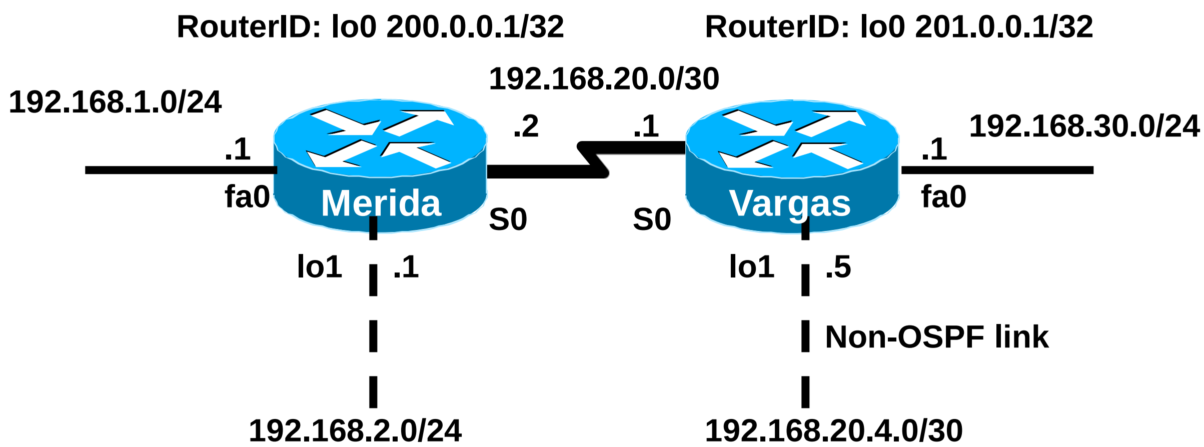
Rtr(config-if)#**ip add 10.5.1.1 255.255.255.0**

Rtr(config)# **router ospf 10**

Rtr(config-router)#**network 10.5.1.0 0.0.0.255 area 0**

Network Command and the Wildcard Mask

Cabrillo College



Merida

```
Merida(config)#router ospf 1
Merida(config-router)#network 192.168.1.0 0.0.0.255 area 0
Merida(config-router)#network 192.168.2.0 0.0.0.255 area 0
Merida(config-router)#network 192.168.20.0 0.0.0.3 area 0
```

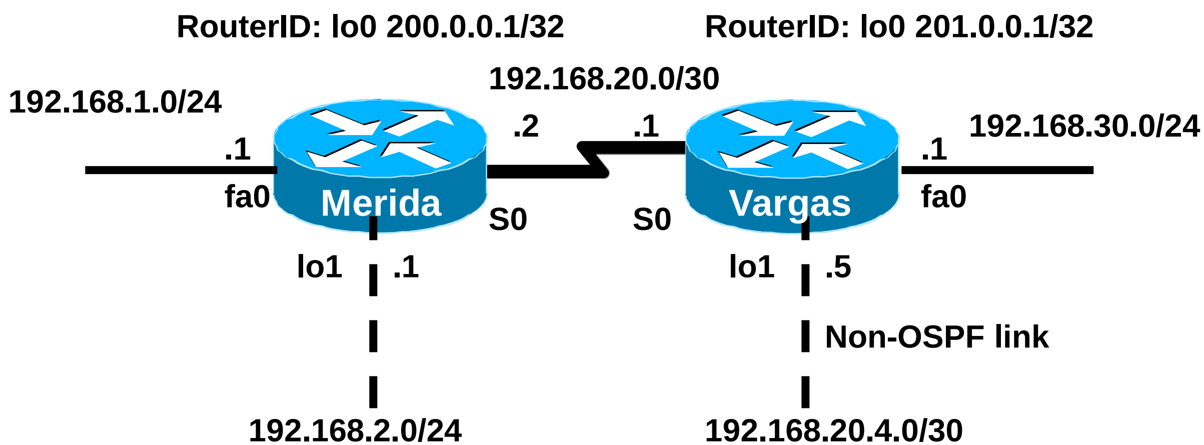
Vargas

```
Vargas(config)#router ospf 10
Vargas(config-router)#network 192.168.20.0 0.0.0.3 area 0
Vargas(config-router)#network 192.168.30.0 0.0.0.255 area 0
```

Only 192.168.20.0/30 255.255.255.252 NOT 192.168.20.4/30

Network Command and the Wildcard Mask

Cabrillo College



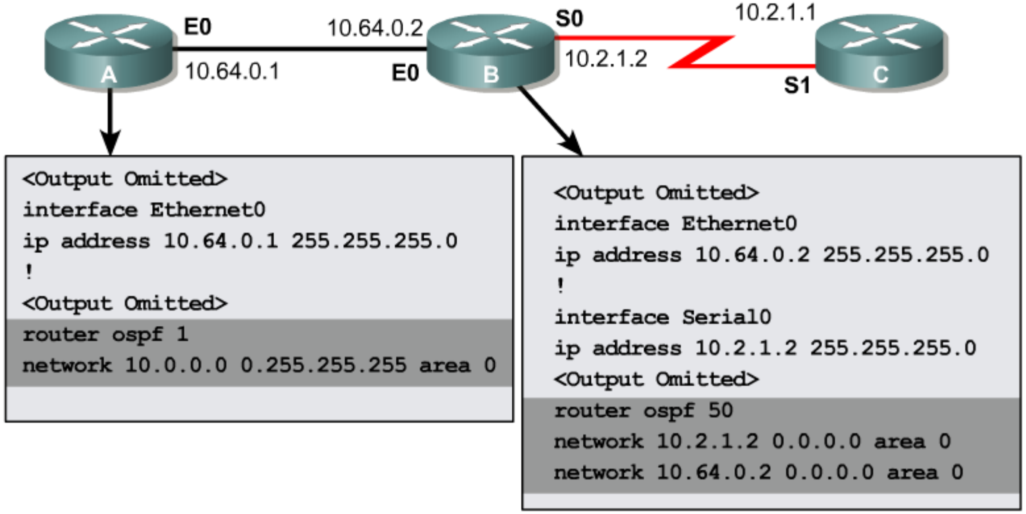
First three octets of the address must match 192.168.3.0 0.0.0.3

- Last octet of the network address is 0 = 00000000
- Last octet of the wildcard mask address is 3 = 00000011
- Must match the first 6 bits of the address 000000
- Don't care about the last two bits of the address 11
- 'Addresses' that would match 00000000, 00000001, 00000010, 00000011
  - 192.168.20.0, 192.168.20.1, 192.168.20.2, 192.168.20.3
- 'Address' that does NOT match: 00000101 or 192.168.20.5

Only 192.168.20.0/30 NOT 192.168.20.4/30

```
Vargas(config-router)#network 192.168.20.0 0.0.0.3 area 0
```

Configuring the Network Command - Extra



Other times you may wish to get more specific or less specific.

```
Rtr(config-if)#ip add 10.5.1.1 255.255.255.0
```

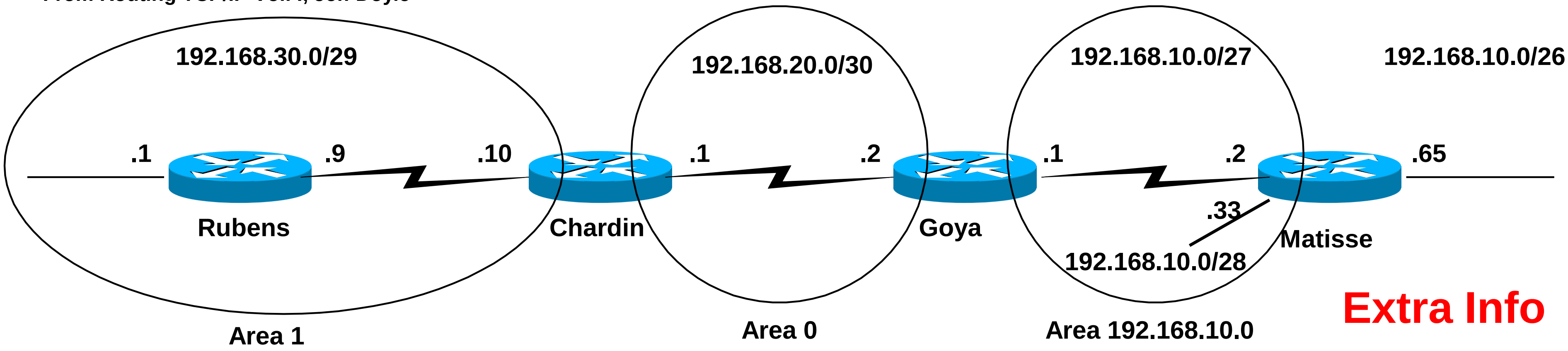
```
Rtr(config)# router ospf 10
Rtr(config-router)#network 0.0.0.0 255.255.255.255 area 0
```

- Matches all interfaces on this router, not recommended

```
Rtr(config)# router ospf 10
Rtr(config-router)#network 10.5.1.2 0.0.0.0 area 0
```

- Matches only the interface 10.5.1.2 and not any other 10.5.1.n interfaces.

From Routing TCP/IP Vol. I, Jeff Doyle



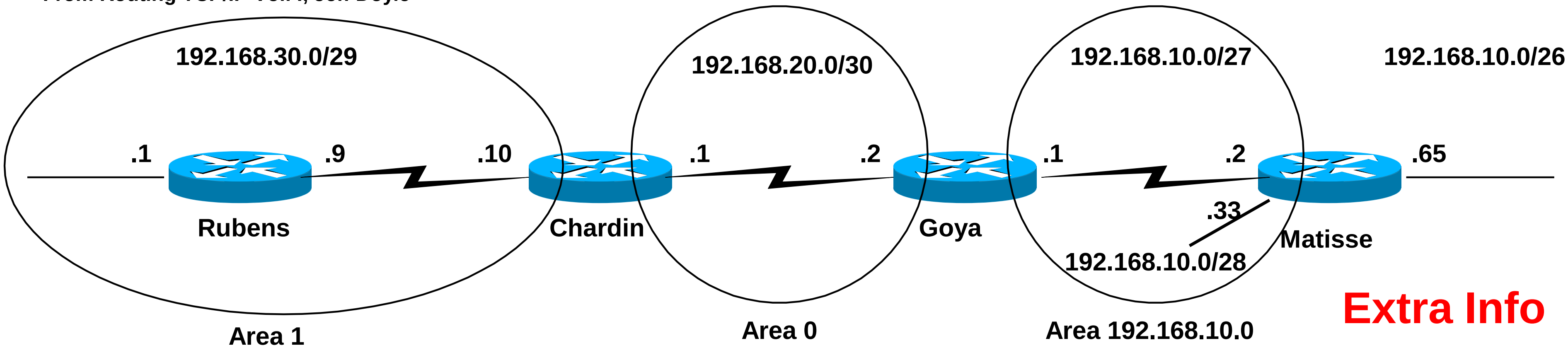
Extra Info

Rubens

```
router ospf 10
   network 0.0.0.0 255.255.255.255 area 1
```

- This will match all interfaces on the router.
- The address 0.0.0.0 is just a placeholder, the inverse mask of 255.255.255.255 does the actual matching with "don't care" bits placed across the entire four octets of the address.
- This method provides the least precision control and is generally discouraged against, as you may bring up another interface on the router and you did not mean to run OSPF on that interface.

From Routing TCP/IP Vol. I, Jeff Doyle



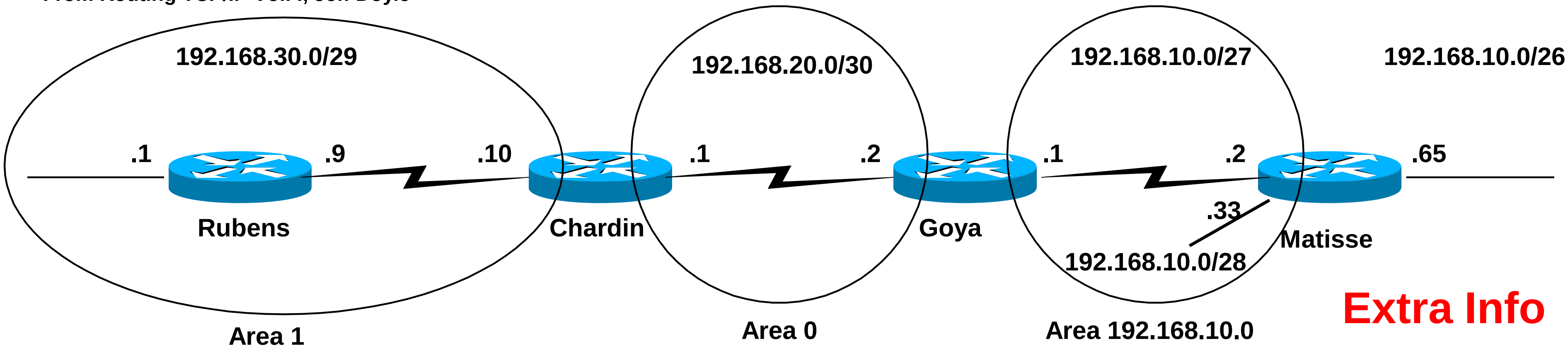
Extra Info

Chardin

```
router ospf 20
   network 192.168.30.0 0.0.0.255 area 1
   network 192.168.20.0 0.0.0.255 area 0
```

- Chardin is a ABR (Area Border Router) which we will discuss next chapter, and belongs to two different areas.
- We need to be more specific here as each interface belongs to a different area.
- Here we are saying that any interface that has 192.168.30.n in the first three octets belongs to area 1 and any interface that has 192.168.20.n in the first three octets belongs to area 0.
- Notice that the inverse mask does <u>not</u> have to inversely match the subnet mask of the interface (255.255.255.248 and 255.255.255.252).

From Routing TCP/IP Vol. I, Jeff Doyle



Extra Info

Goya

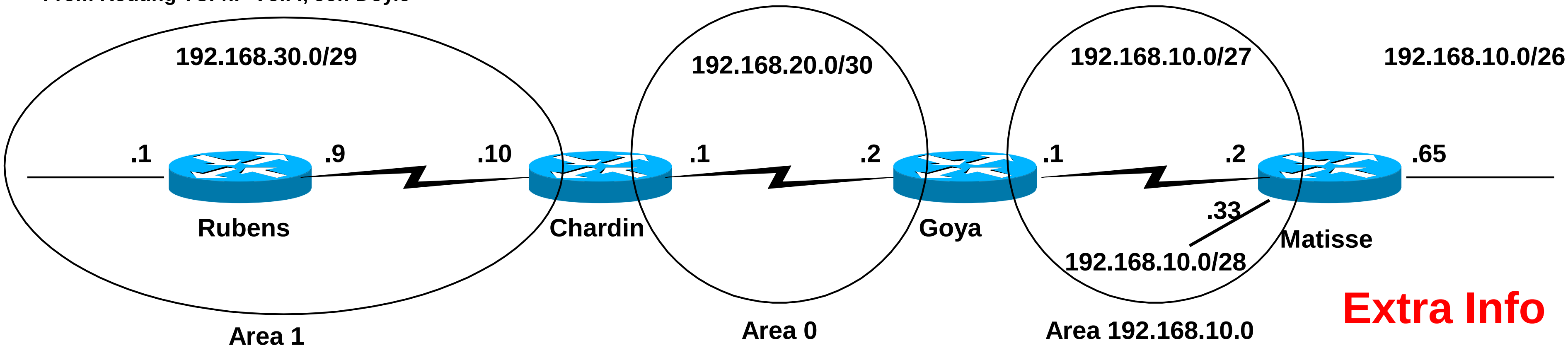
```
router ospf 30
   network 192.168.20.0 0.0.0.3 area 0.0.0.0
   network 192.168.10.0 0.0.0.31 area 192.168.10.0
```

- Goya is also an ABR.
- The network statements will only match the specific subnets configured on the two interfaces.

```
/30 = 255.255.255.252 = 11111100  00 = host bits
3  = 00000011  - Match last two bits of subnet mask

/27 = 255.255.255.224 = 11100000  00000 = host bits
31 = 00011111  - Match last five bits of subnet mask
```

From Routing TCP/IP Vol. I, Jeff Doyle



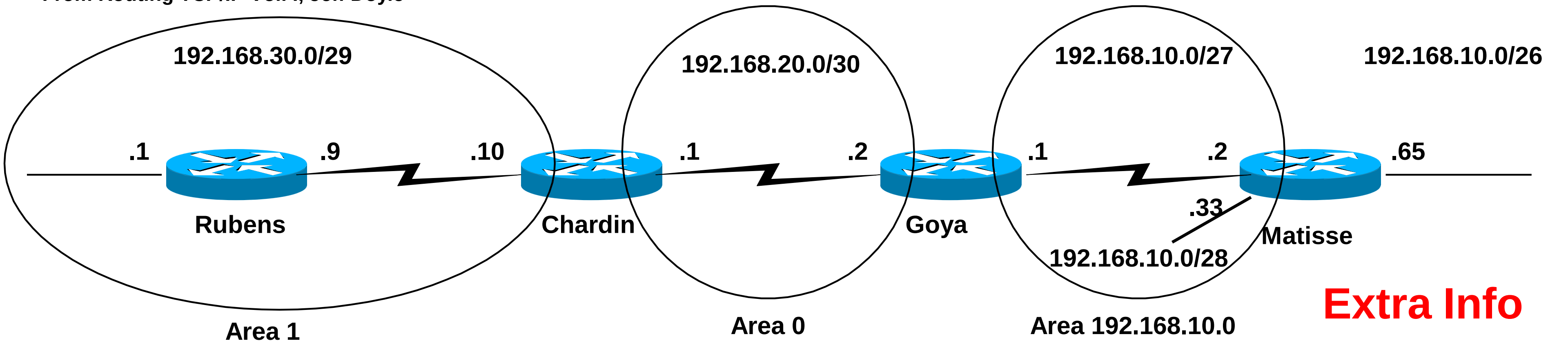
Extra Info

Goya

```
router ospf 30
   network 192.168.20.0 0.0.0.3 area 0.0.0.0
   network 192.168.10.0 0.0.0.31 area 192.168.10.0
```

- Goya is also an ABR.
- Also notice that you can use an dotted decimal notation to represent an area.
- In my experience it is not very common, but when it is used, most people use the network address.
- Area 0 can be represented as 0 or 0.0.0.0.
  - When the dotted decimal is used OSPF packets are converted to “0” so the two can be compatible.

From Routing TCP/IP Vol. I, Jeff Doyle



Extra Info

Matisse

```
router ospf 40
   network 192.168.10.2 0.0.0.0 area 192.168.10.0
   network 192.168.10.33 0.0.0.0 area 192.168.10.0
```

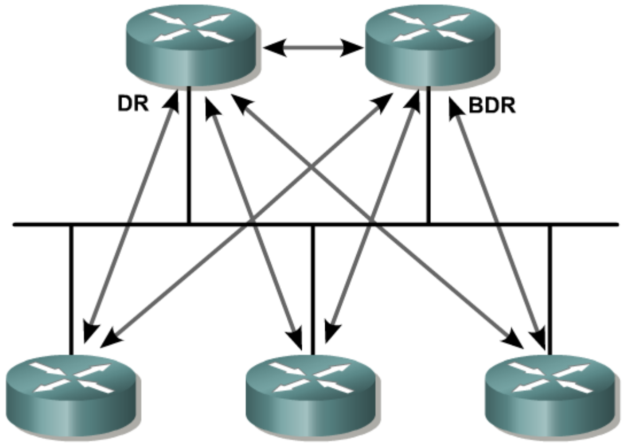
- Matisse has one interface, 192,168,10.65/26, which is <u>not</u> running OSPF.
- The network statements for this router are configured specifically for the individual addresses and the inverse mask indicates that all 32 bits must match exactly.
- This method provides the most precise control over which interfaces will run OSPF.

Configuring a Loopback Address


```
Rtr(config)# interface loopback 0
Rtr(config-if)# ip add 10.1.1.1 255.255.255.0
```

- Automatically are “up” and “up”
- Very useful in setting Router IDs as they never go down.
- RouterID is used to identify the routers in the OSPF network
  - IP address configured with the Router-ID command (extra)
  - Highest loopback address
  - Highest active IP address
- Important for DR/BDR elections unless you use the ip ospf priority command (next)
- Extra: Also, useful to configure “virtual” networks that you can ping and route as if they were attached networks.

DR/BDR Elections



- Router with the highest Router ID is elected the DR, next is BDR.
- But like other elections, this one can be rigged.

```
Rtr(config)# interface fastethernet 0
Rtr(config-if)# ip ospf priority <0-255>
```

- Higher priority becomes DR/BDR
- Default = 1
- Ineligible to become DR/BDR = 0

Modifying the Cost

`Rtr(config-if)#` **`bandwidth 64`** **=** `Rtr(config-if)#` **`ip ospf cost 1562`**

`bandwidth` **command**

`Rtr(config-if)#` **`bandwidth`** ***`kilobits`***

`(ex: 64 = 64,000bps)`

- Changes the default bandwidth metric on a specific interface.
- Used in the 10^8/bandwidth calculation for cumulating the cost of a route from the router to the network on the outgoing interfaces.
- Does not modify the actual speed of the link.

`ip ospf cost` **command**

`RTB(config-if)#` **`ip ospf cost`** ***`value`***

`(ex: 1562, same as bandwidth = 64kbps)`

- Configures the cost metric for a specific interface
- Uses this value for the cost of this interface instead of the 10^8/bandwidth calculation
- Common for multivendor environments.

Configuring Simple Authentication

A router, by default, trusts that routing information received, has come from a router that should be sending it.

`Rtr(config-if)#` **`ip ospf authentication-key`** ***`passwd`***

- Configured on an interface
- *password* = Clear text unless message-digest is used (next)
  - Easily captured using a packet sniffer
  - Passwords do <u>not</u> have to be the same throughout an area, but they must be same between neighbors.

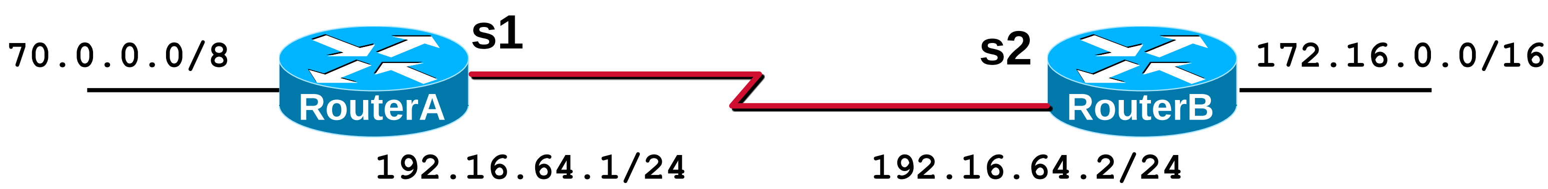
After a password is configured, you enable authentication for the area on all participating area routers with:

`Rtr(config-router)#` **`area`** ***`area`*** **`authentication`**

- Configured for an OSPF area, in ospf router mode.

Configuring Simple Authentication

Cabrillo College



RouterA

```
interface Serial1
  ip address 192.16.64.1 255.255.255.0
  ip ospf authentication-key secret
  !
router ospf 10
  network 192.16.64.0 0.0.0.255 area 0
  network 70.0.0.0 0.255.255.255 area 0
  area 0 authentication
```

RouterB

```
interface Serial2
 ip address 192.16.64.2 255.255.255.0
 ip ospf authentication-key secret
!
router ospf 10
 network 172.16.0.0 0.0.255.255 area 0
 network 192.16.64.0 0.0.0.255 area 0
 area 0 authentication
```

Configuring MD5 Encrypted Authentication

Rtr(config-if)# **ip ospf message-digest-key *key-id* md5 *password***

- *Key-id* = 1 to 255, must match on each router to authenticate.
- *md5* = Encryption-type
- *password* = encrypted
  - –Passwords do <u>not</u> have to be the same throughout an area, but they must be same between neighbors.

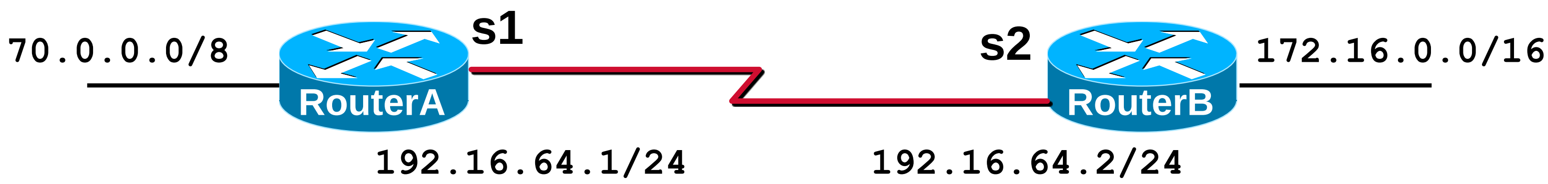
After a password is configured, you enable authentication for the area on all participating area routers with:

Rtr(config-router)# **area *area* authentication [message-digest]**

- message-digest option must be used if using message-digest-key
- If optional message-digest is used, a message digest, or hash, of the password is sent.

Configuring MD5 Encrypted Authentication

Cabrillo College




```
RouterA
interface Serial1
  ip address 192.16.64.1 255.255.255.0
  ip ospf message-digest-key 1 md5 secret
  !
router ospf 10
  network 192.16.64.0 0.0.0.255 area 0
  network 70.0.0.0 0.255.255.255 area 0
  area 0 authentication message-digest
```

```
RouterB
interface Serial2
 ip address 192.16.64.2 255.255.255.0
 ip ospf message-digest-key 1 md5 secret
!
router ospf 10
 network 172.16.0.0 0.0.255.255 area 0
 network 192.16.64.0 0.0.0.255 area 0
 area 0 authentication message-digest
```

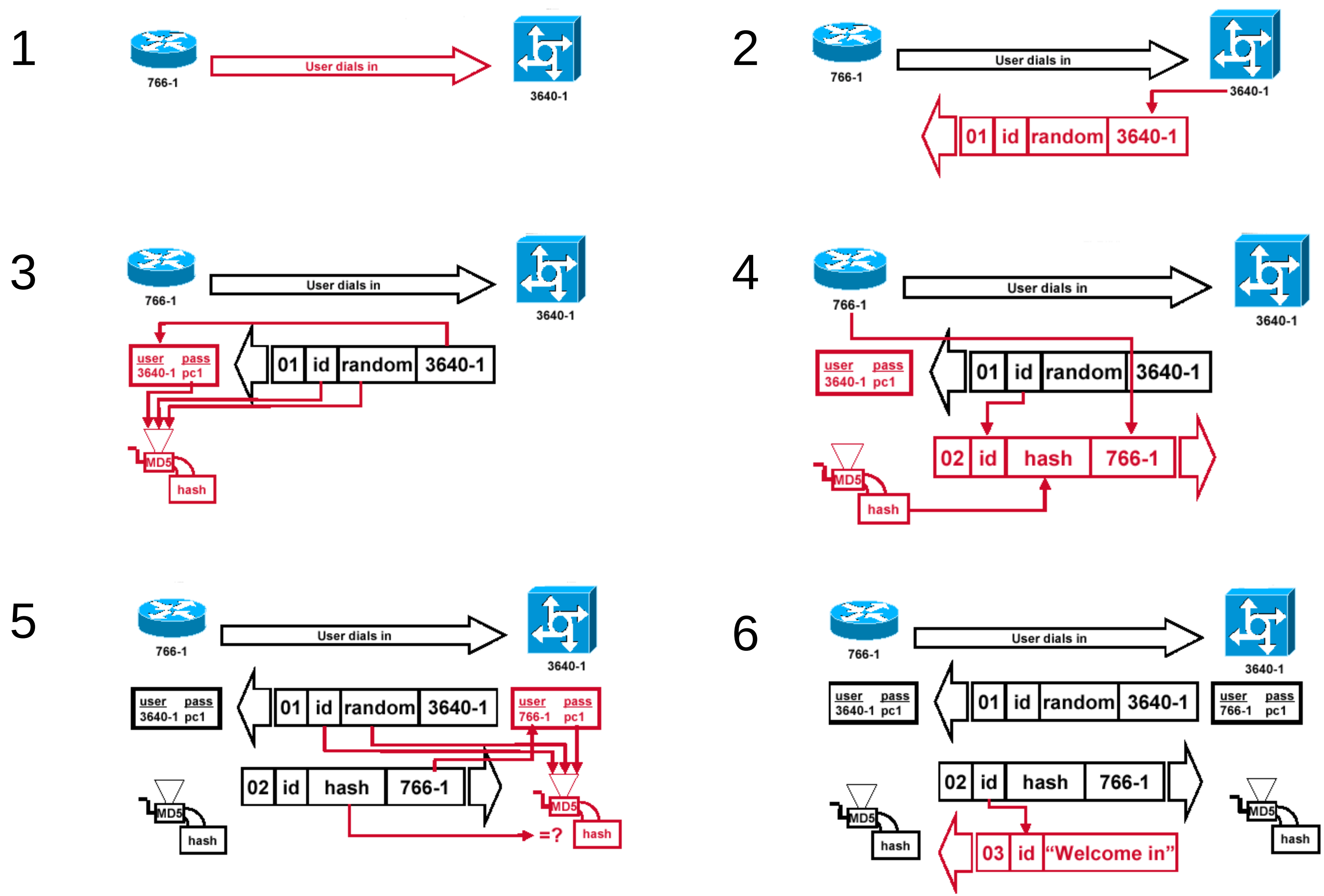
MD5 Encryption

- MD5 authentication, creates a message digest.
- This is scrambled data that is based on the password and the packet contents .
- The receiving router uses the shared password and the packet to re-calculate the digest.
- If the digests match, the router believes that the source of the packet and its contents have not been tampered with.
- In the case of message-digest authentication, the authentication data field contains the key-id and the length of the message digest that is appended to the packet.
- The Message Digest is like a watermark that can’t be faked.

MD5 Authentication (FYI)

http://www.cisco.com/en/US/tech/tk713/tk507/technologies_tech_note09186a00800b4131.shtml

Cabrillo College



Configuring OSPF Timers


```
Rtr(config-if)# ip ospf hello-interval seconds
Rtr(config-if)# ip ospf dead-interval seconds
```

- Configured on an interface
- For OSPF routers to be able to exchange information, the must have the same hello intervals and dead intervals.
- By default, the dead interval is 4 times the hello interval, so the a router has four chances to send a hello packet being declared dead. (not required)
- In multi-vendor networks, Hello timers may need to be adjusted.
- Do not modify defaults unless you have a compelling need to do so.

Defaults

- On broadcast networks hello interval = 10 seconds, dead interval 40 seconds.
- On non-broadcast networks hello interval = 30 seconds, dead interval 120 seconds.
- **Note**: On some IOS's, the deadinterval automatically changes when the hellointerval is modified.

Configuring and Propagating a Default Route

Cabrillo College

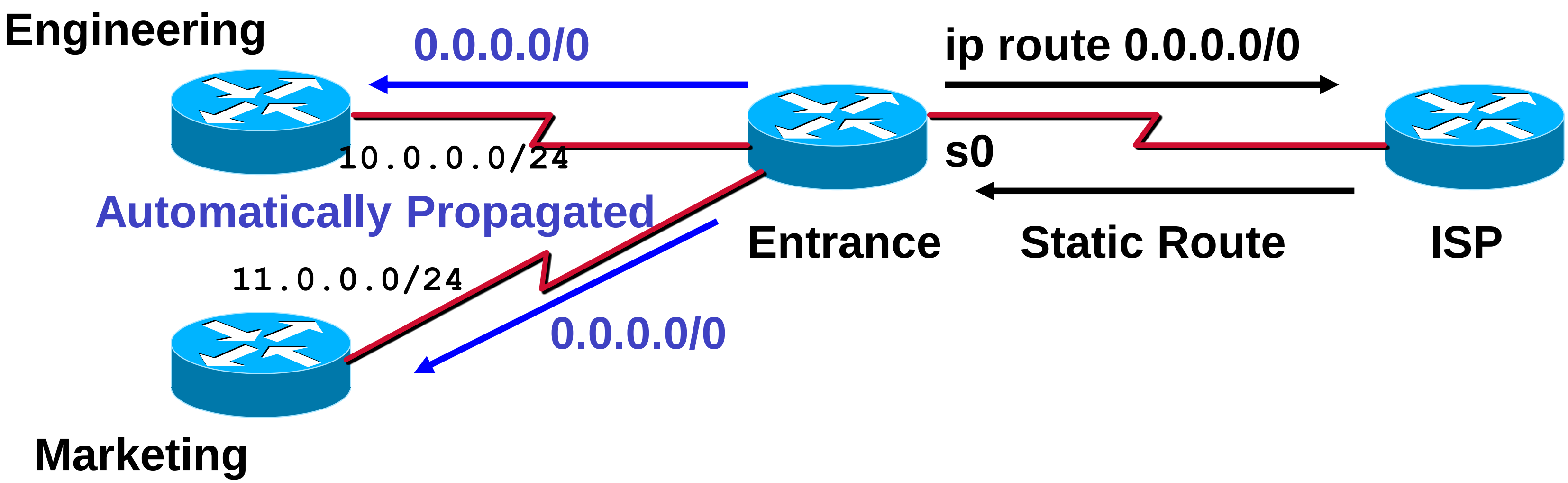
```
Router(config)# ip route 0.0.0.0 0.0.0.0 serial0

Router(config)# router ospf 1
Router(config-router)# default-information originate [always]
```

- If the ASBR has a default route configured (ip route 0.0.0.0 0.0.0.0), the default-information originate command is necessary to advertise 0.0.0.0/0 to the other routers in the area.
- If the default-information originate command is <u>not</u> used, the default “quad-zero” route will <u>not</u> be propagated.
- *Important*: The default route and the default-information originate command are usually only be configured on your “Entrance” or “Gateway” router, the router that connects your network to the outside world.
  - –This router is known as the ASBR (Autonomous System Boundary Router)
- *Extra*: The always option will propagate a default “quad-zero” route even if one is not configured on this router.

Default Route Example

Cabrillo College



Engineering and Marketing will have 0.0.0.0/0 default routes forwarding packets to the Entrance router.

```
Entrance(config)# ip route 0.0.0.0 0.0.0.0 serial 0
Entrance(config)# router ospf 1
Entrance(config-router)# network 10.0.0.0 0.0.0.255 area 0
Entrance(config-router)# network 11.0.0.0 0.0.0.255 area 0
Entrance(config-router)# default-information originate
```

show ip route


```
Router# show ip route
  172.16.0.0/16 is variably subnetted, 4 subnets, 3 masks
O IA    172.16.51.1/32 [110/783] via 172.16.1.2, 00:11:44,
   FastEthernet0
O       172.16.20.0/24 [110/782] via 172.16.10.6, 00:12:29, Serial0
C       172.16.10.4/30 is directly connected, Serial0
C       172.16.1.0/24 is directly connected, FastEthernet0
O E2 11.0.0.0/8 [110/20] via 172.16.1.1, 00:11:44, FastEthernet0
O E1 12.0.0.0/8 [110/782] via 172.16.1.1, 00:11:44, FastEthernet0
```

- **`O`** = OSPF routes within the same area (intra-area routes)
- **`110/number`** = Administrative Distance/metric (cumulative 10^8/bandwidth)
- **`E2`** = Routes outside of the OSPF routing domain, redistributed into OSPF.
  - Default is E2 with a cost of **20** and does **<u>not</u>** get modified within the OSPF
- **`O IA`** = OSPF routes from another area (inter-area routes)
- **`E1`** = Routes outside of the OSPF routing domain and get additional cumulative costs added on by each router, just like other OSPF routes.

show ip ospf

Cabrillo College

```
Router#show ip ospf
 Routing Process "ospf 1" with ID 192.168.3.1
 Supports only single TOS(TOS0) routes
 It is an area border router
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
 Number of external LSA 3. Checksum Sum 0x97E3
 Number of DCbitless external LSA 0
 Number of DoNotAge external LSA 0
 Number of areas in this router is 2. 2 normal 0 stub 0 nssa
 External flood list length 0
    Area BACKBONE(0)
      Number of interfaces in this area is 1
      Area has no authentication
      SPF algorithm executed 8 times
      <text omitted>
    Area 1
       <text omitted>
```

show ip ospf interface


```
Router# show ip ospf interface
Ethernet0 is up, line protocol is up
  Internet Address 206.202.2.1/24, Area 1
  Process ID 1, Router ID 1.2.202.206, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 2.2.202.206, Interface address 206.202.2.2
  Backup Designated router (ID) 1.2.202.206, Interface address 206.202.2.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:00
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 2.2.202.206  (Designated Router)
  Suppress hello for 0 neighbor(s)
Serial0 is up, line protocol is up
  Internet Address 206.202.1.2/24, Area 1
  Process ID 1, Router ID 1.2.202.206, Network Type POINT_TO_POINT, Cost:
   64
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:04
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 2.0.202.206
  Suppress hello for 0 neighbor(s)
```

show ip ospf neighbor

Cabrillo College

```
RouterB#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
1.5.202.206       1   FULL/DROTHER    00:00:33    206.202.0.3     Ethernet0
1.10.202.206      1   FULL/BDR        00:00:32    206.202.0.4     Ethernet0
1.0.202.206       1   2WAY/DROTHER    00:00:30    206.202.0.1     Ethernet0
1.2.202.206       1   FULL/  -        00:00:32    206.202.1.2     Serial0
```

- In this example, we are the DR
- DROTHER may be in FULL or 2 WAY state, both cases are normal.
- Usually if there are multiple DROTHERs, they will be in either FULL or 2WAY state but not both.

debug ip ospf adj (adjacency)


```
Router# debug ip ospf adj
04:19:46: OSPF: Rcv hello from 201.0.0.1 area 0 from FastEthernet0 192.168.20.1
04:19:46: OSPF: 2 Way Communication to 201.0.0.1 on FastEthernet0, state 2WAY
04:19:46: OSPF: End of hello processing
<text omitted>
04:20:22: OSPF: end of Wait on interface FastEthernet0
04:20:22: OSPF: DR/BDR election on FastEthernet0
04:20:22: OSPF: Elect BDR 200.0.0.1
04:20:22: OSPF: Elect DR 200.0.0.1
04:20:22: OSPF: Elect BDR 201.0.0.1
04:20:22: OSPF: Elect DR 200.0.0.1
04:20:22:        DR: 201.0.0.1 (Id)   BDR: 200.0.0.1 (Id)
04:20:23: OSPF: Rcv DBD from 201.0.0.1 on FastEthernet0 seq 0x2657 opt 0x2 flag
0x7 len 32  mtu 1500 state EXSTART
04:20:23: OSPF: NBR Negotiation Done. We are the SLAVE
04:20:23: OSPF: Send DBD to 201.0.0.1 on FastEthernet0 seq 0x2657 opt 0x2 flag 0 x2 len 92
04:20:23: OSPF: Rcv DBD from 201.0.0.1 on FastEthernet0 seq 0x2658 opt 0x2 flag
0x3 len 72  mtu 1500 state EXCHANGE
<text omitted>
04:20:23: OSPF: Synchronized with 201.0.0.1 on FastEthernet0, state FULL
```

- **Displays adjacency information including Hello processing, DR/BDR election, authentication, and the "Steps to OSPF Operation."**

debug ip ospf events


```
Router# debug ip ospf events
08:00:56: OSPF: Rcv hello from 201.0.0.1 area 0 from FastEthernet0 192.168.20.1
08:00:56: OSPF: Mismatched hello parameters from 192.168.20.1
08:00:56: Dead R 40 C 20, Hello R 10 C 5  Mask R 255.255.255.252 C
   255.255.255.2
52
```

- Shows much of the same information as debug ip ospf adj in the previous slide including, adjacencies, flooding information, designated router selection, and shortest path first (SPF) calculation.
- This information is also displayed with debug ip ospf events.
- R = Received
- C = Current (?)

Later: show ip ospf database (summary of link state database)


```
Internal#show ip ospf data
        OSPF Router with ID (192.168.4.1) (Process ID 1)
```

Link states within this area, this is what the SPF uses.

```
        Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
192.168.3.1     192.168.3.1     898         0x80000003 0xCE56   2
192.168.4.1     192.168.4.1     937         0x80000003 0xFD44   3
```

Link states of any DRs in this area.

```
        Summary Net Link States (Area 0)

 Link ID         ADV Router      Age         Seq#       Checksum
172.16.1.0      192.168.3.1     848         0x80000005 0xD339
172.16.51.1     192.168.3.1     843         0x80000001 0xB329
```

Link states summaries of links outside this area. (No SPF)

```
        Summary ASB Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
192.168.1.1     192.168.3.1     912         0x80000003 0x93CC
```

Link states summaries of links external routes. (No SPF)

```
        Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
11.0.0.0        192.168.1.1     1302        0x80000001 0x3FEA   0
12.0.0.0        192.168.1.1     1303        0x80000001 0x32F6   0
```

OSPF Configuration Commands - Review

Required Commands:

Rtr(config)# **router ospf *process-id***

Rtr(config-router)#**network *address wildcard-mask* area *area-id***

Optional Commands:

Rtr(config-router)# **default-information originate** **(Send default)**

Rtr(config-router)# **area *area* authentication** **(Plain authen.)**

Rtr(config-router)# **area *area* authentication message-digest**

(md5 authen.)

Rtr(config)# **interface loopback *number*** **(Configure lo as RtrID)**

Rtr(config)# **interface type slot/port**

Rtr(config-if)# **ip ospf priority <0-255>** **(DR/BDR election)**

Rtr(config-if)# **bandwidth *kbps*** **(Modify default bandwdth)**

RTB(config-if)# **ip ospf cost *cost*** **(Modify inter. cost)**

Rtr(config-if)# **ip ospf hello-interval *seconds*** **(Modify Hello)**

Rtr(config-if)# **ip ospf dead-interval *seconds*** **(Modify Dead)**

Rtr(config-if)# **ip ospf authentication-key *passwd*** **(Plain/md5authen)**

Rtr(config-if)# **ip ospf message-digest-key *key-id* md5 *password***

OSPF Show Commands - Review

Cabrillo College

```
Router# show ip route
Router# show ip ospf
Router# show ip ospf interface
Router# show ip ospf neighbor
Router# show ip ospf database
Router# debug ip ospf adj
Router# debug ip ospf events
```

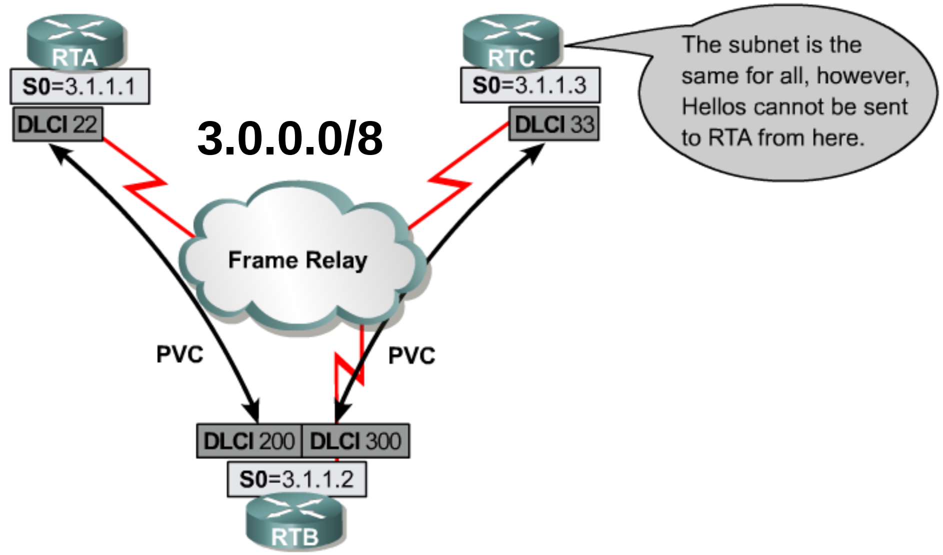
Configuring OSPF over NBMA

NBMA Overview

Network Type	Determined Characteristics	DR Election?
Broadcast multiaccess	Ethernet, Token Ring, or FDDI	Yes
Nonbroadcast multicast	Frame Relay, X.25, SMDS	Yes
Point-to-point	PPP, HDLC	No
Point-to-multipoint	Configured by an administrator	No

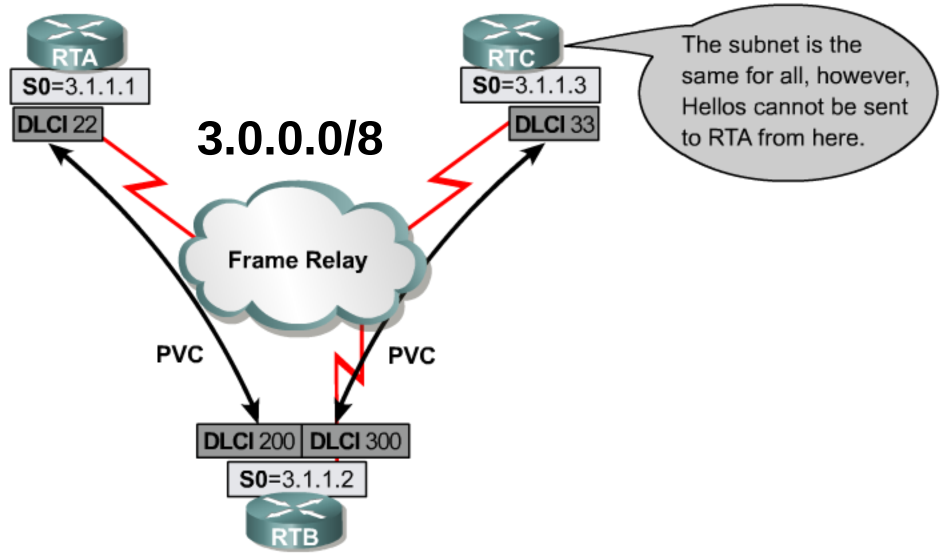
- NBMA networks can create problems with OSPF operation, specifically with the exchange of multicast Hello packets.
- NonBroadcast routers that belong to the same IP subnetwork and will attempt to elect a DR and a BDR.
- However, these routers cannot hold a valid election if they cannot receive multicast Hellos from every other router on the network. Without administrative intervention, a strange election takes place. As far as RTA is concerned, RTC is not participating. Likewise, RTC goes through the election process oblivious to RTA. This botched election can lead to problems if the central router, RTB, is not elected the DR.

NBMA Overview



- NonBroadcast routers that belong to the same IP subnetwork and will attempt to elect a DR and a BDR.
- However, these routers cannot hold a valid election if they cannot receive multicast Hellos from every other router on the network. Without administrative intervention, a strange election takes place.
- As far as RTA is concerned, RTC is not participating.
- Likewise, RTC goes through the election process oblivious to RTA.
- This botched election can lead to problems if the central router, RTB, is not elected the DR.

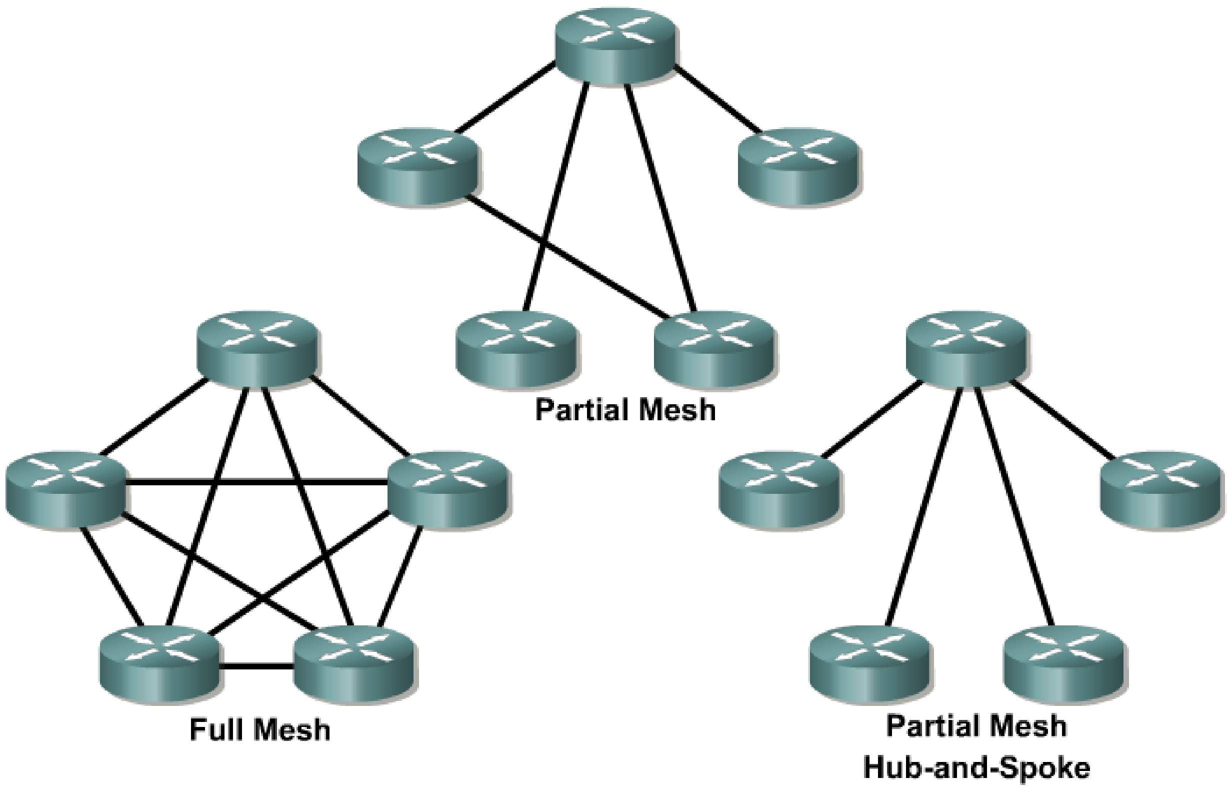
NBMA Overview



NBMA networks support multiple nodes on the same IP network without fully supporting broadcasts and multicasts.

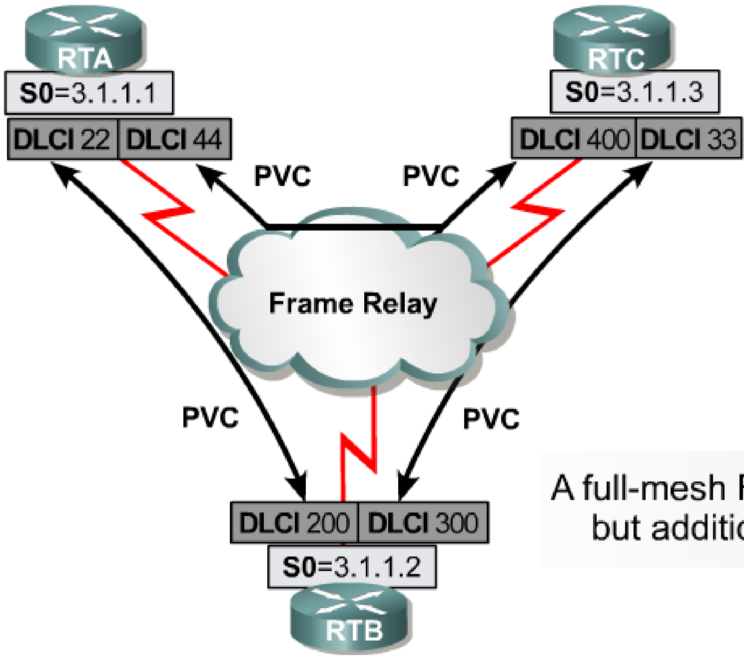
- The Cisco IOS offers several options for configuring OSPF to overcome NBMA limitations, including the OSPF `neighbor` command, point-to-point subinterfaces, and point-to-multipoint configuration.
- The solutions that are available depend on the current NBMA network topology.

Full Mesh Frame Relay



The full-mesh topology requires the most PVCs, and a hub-and-spoke topology requires the fewest. The hub-and-spoke topology is a specific kind of partial-mesh topology.

Full Mesh Frame Relay – Physical Interfaces

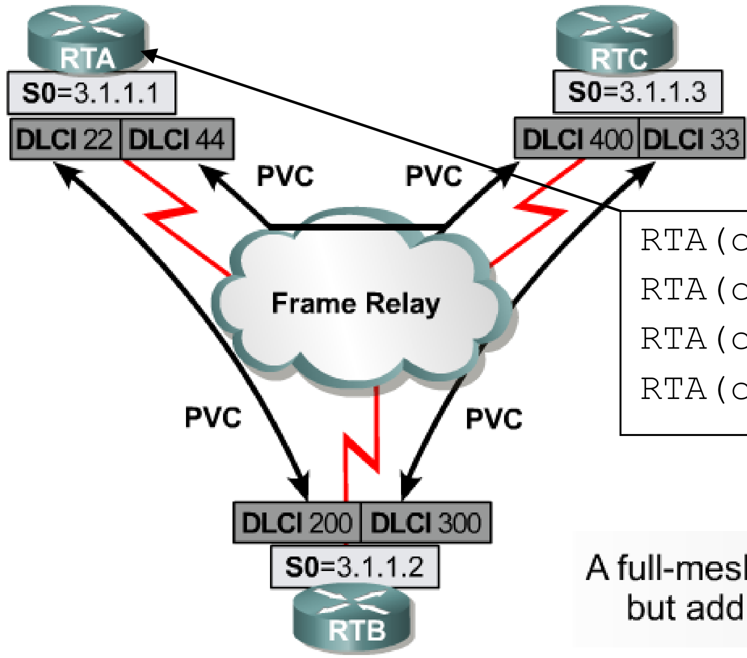


Nonbroadcast interfaces (DR/BDR)

A full-mesh Frame Relay can connect multiple nodes on the same IP subnet, but additional configuration is neccessary to make OSPF work properly.

- Organizations use Frame Relay primarily because it supports more than one logical connection over a single interface.
- This makes it an affordable and flexible choice for WAN links.
- A full-mesh topology takes advantage of the capabilities Frame Relay has to support multiple permanent virtual circuits (PVCs) on a single serial interface.
- In a full-mesh topology, every router has PVCs to every other router.

Full Mesh Frame Relay – Physical Interfaces



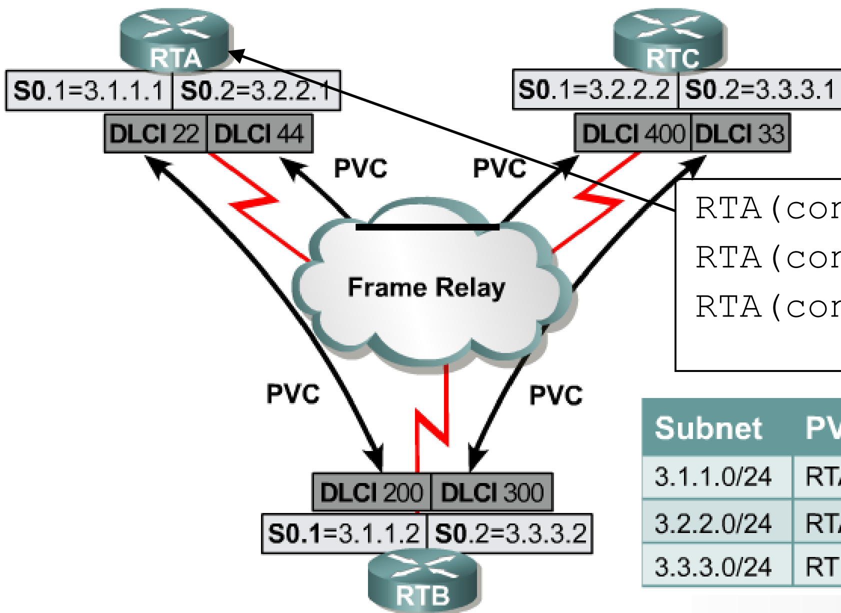
Nonbroadcast interfaces (DR/BDR)

```
RTA(config)#router ospf 1
RTA(config-router)#network 3.1.1.0 0.0.0.255 area 0
RTA(config-router)#neighbor 3.1.1.2
RTA(config-router)#neighbor 3.1.1.3
```

A full-mesh Frame Relay can connect multiple nodes on the same IP subnet, but additional configuration is neccessary to make OSPF work properly.

- For OSPF to work properly over a multiaccess full-mesh topology that does not support broadcasts, each OSPF neighbor addresses must be manually entered on each router, one at a time.
- The OSPF **`neighbor`** command tells a router about the IP addresses of its neighbors so that it can exchange routing information without multicasts.
- Specifying the neighbors for each router is not the only option to make OSPF work in this type of environment.
- The following section explains how configuring subinterfaces can eliminate the need for the **`neighbor`** command.

Full Mesh Frame Relay – Subinterfaces



Point-to-point interfaces (No DR/BDR)

```
RTA(config)#router ospf 1
RTA(config-router)#network 3.1.1.0 0.0.0.255 area 0
RTA(config-router)#network 3.2.2.0 0.0.0.255 area 0
```

Subnet	PVC
3.1.1.0/24	RTA-RTB
3.2.2.0/24	RTA-RTC
3.3.3.0/24	RTB-RTC

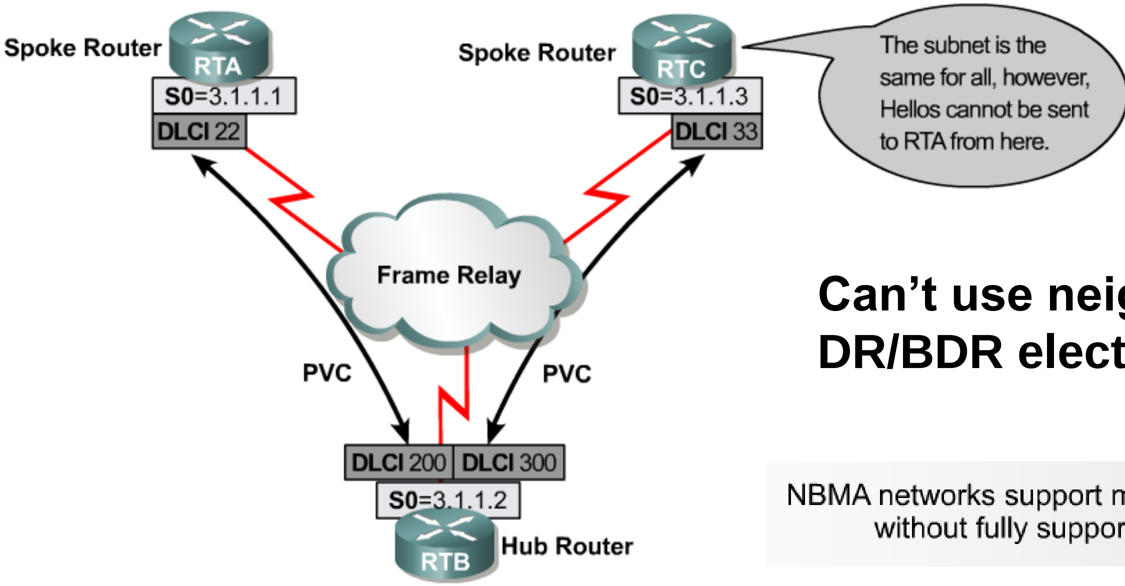
Using subinterfaces, a full-mesh Frame Relay can be logically separated into point-to-point networks, each with their own IP subnet.

- The IOS subinterface feature can be used to break up a multiaccess network into a **collection of point-to-point networks**.
- OSPF automatically recognizes this configuration as point-to-point, not NBMA, even with Frame Relay configured on the interfaces.
- Recall that OSPF point-to-point networks **do not elect a DR**.
- Instead, the Frame Relay router uses Inverse ARP or a Frame Relay map to obtain the link partner's address so that routing information can be exchanged.

Full Mesh Frame Relay

- A full-mesh topology offers numerous advantages, including maximum fault tolerance.
- Unfortunately, full-mesh topologies can get expensive because each PVC must be leased from a provider.
- An organization would have to lease 45 PVCs to support just ten fully meshed routers.
- If subinterfaces are used to create point-to-point networks, then the 45 IP subnets must also be allocated and managed, which is an additional expense.

Hub and Spoke



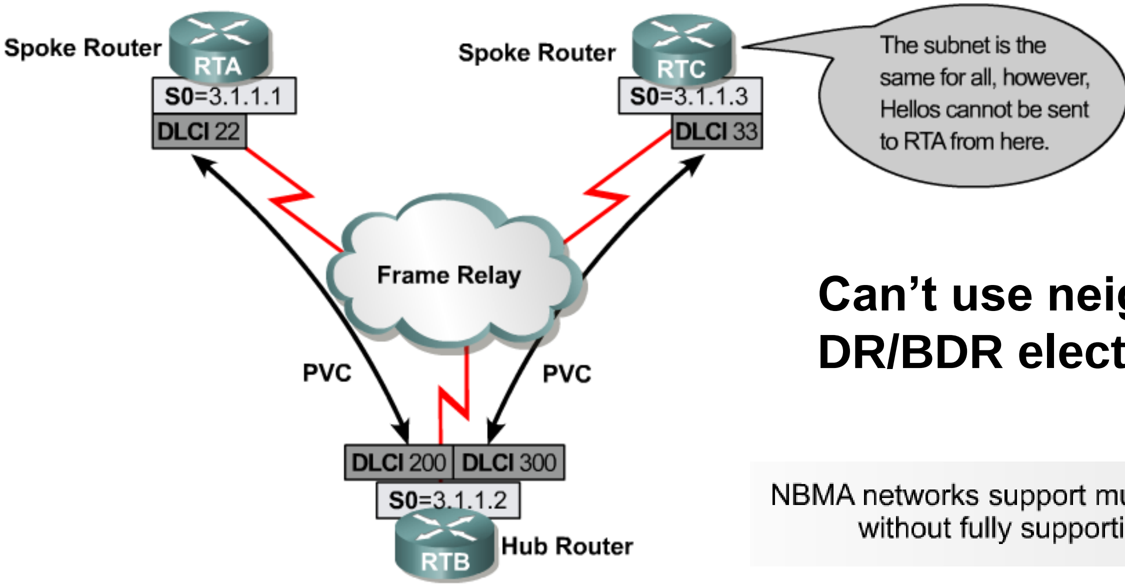
Nonbroadcast interfaces (DR/BDR) – Doesn't work

Can't use neighbor statements for DR/BDR election: not a full mesh.

NBMA networks support multiple nodes on the same IP network without fully supporting broadcasts and multicasts.

- The hub-and-spoke topology is a cost effective WAN solution that introduces a single point of failure, the hub router.
- Organizations typically use Frame Relay because it is inexpensive, not because it is fault tolerant.
- Since dedicated leased lines typically carry mission critical data, an economical Frame Relay topology, such as hub-and-spoke, makes sense.

Hub and Spoke



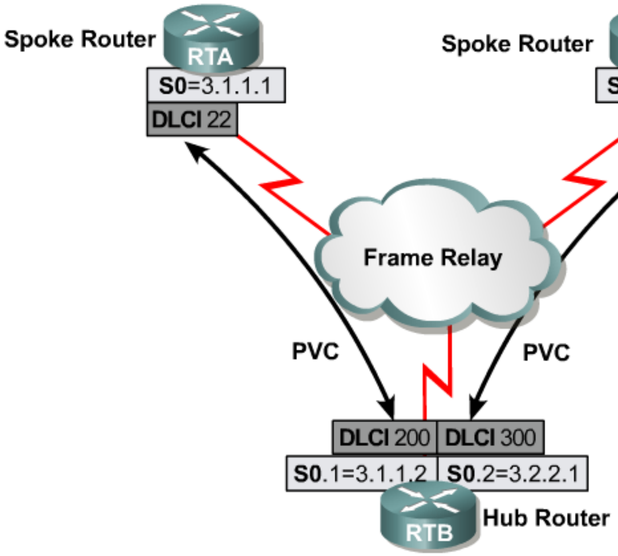
Nonbroadcast interfaces (DR/BDR) – Doesn't work

Can't use neighbor statements for DR/BDR election: not a full mesh.

NBMA networks support multiple nodes on the same IP network without fully supporting broadcasts and multicasts.

- Unfortunately, the **`neighbor`** command that worked with a full-mesh topology does not work as well with the hub-and-spoke topology.
- The hub router sees all the spoke routers and can send routing information to them using the **`neighbor`** command, but the spoke routers can send Hellos only to the hub. (no full mesh PVCs)

Hub and Spoke – Point-to-Point



Point-to-point interfaces (No DR/BDR)

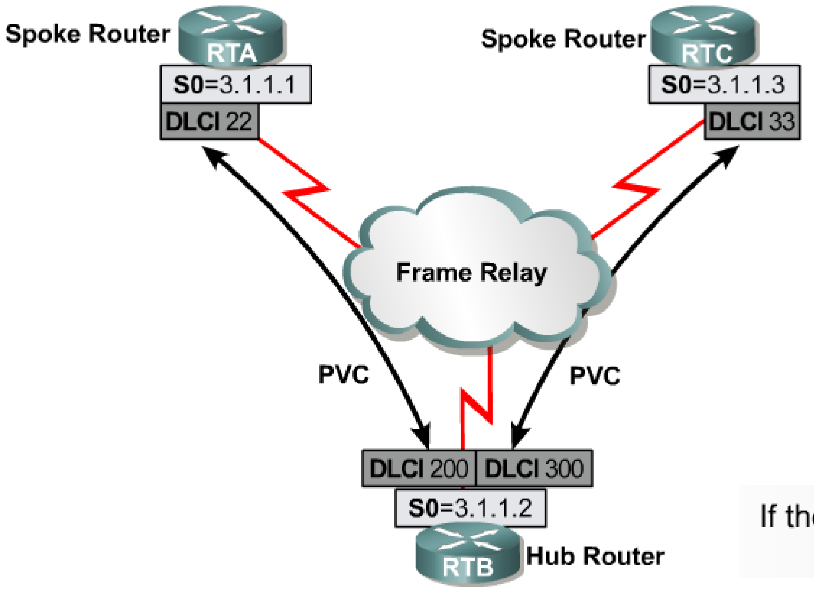
Use point-to-point networks to avoid the DR/BDR election process.

If point-to-point networks are configured in a hub-and-spoke topology, OSPF operation is simplified, but at the cost of using multiple IP subnets.

- To avoid the DR and BDR issue altogether by breaking the network into point-to-point connections. Point-to-point networks will not elect a DR or a BDR.
- Although they make OSPF configuration straightforward, point-to-point networks have major drawbacks when used with a hub-and-spoke topology.
- Subnets must be allocated for each link.
- This can lead to WAN addressing that is complex and difficult to manage.
- The WAN addressing issue can be avoided by using IP unnumbered, but many organizations have WAN-management policies that prevent using this feature. Are there any possible alternatives to a point-to-point configuration?

Hub and Spoke – Point-to-multipoint

Cabrillo College

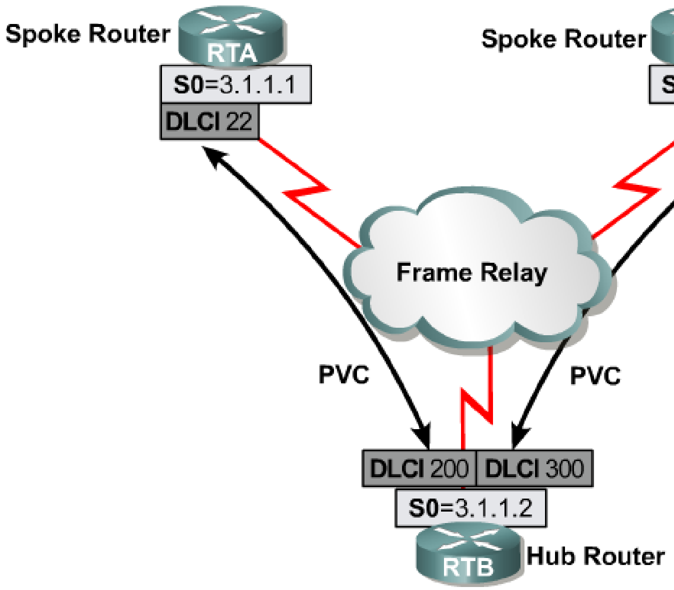


Point-to-multipoint interfaces (No DR/BDR)

If the point-to-multipoint network type is manually configured, routers in a hub-and-spoke topology can exist on the same IP subnet.

- In a point-to-multipoint network, a hub router is directly connected to multiple spoke routers, but all the WAN interfaces are addressed on the same subnet.
- This logical topology was seen earlier in the module.
- However, it was also learned that OSPF does not work properly as an NBMA OSPF network type.
- By manually changing the OSPF network type to point-to-multipoint, this logical topology can then work.

Hub and Spoke – Point-to-multipoint

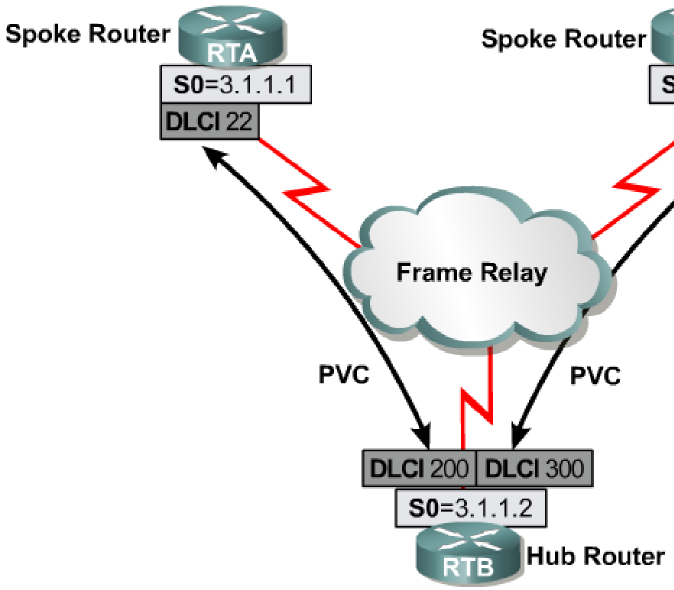


Point-to-multipoint interfaces (No DR/BDR)

If the point-to-multipoint network type is manually configured, routers in a hub-and-spoke topology can exist on the same IP subnet.

- Routing between RTA and RTC will go through the router that has virtual circuits to both routers, RTB.
- Notice that it is not necessary to configure neighbors when using this feature. Inverse ARP will discover them.

Hub and Spoke – Point-to-multipoint



Point-to-multipoint interfaces (No DR/BDR)

If the point-to-multipoint network type is manually configured, routers in a hub-and-spoke topology can exist on the same IP subnet.

Point-to-multipoint networks have the following properties:

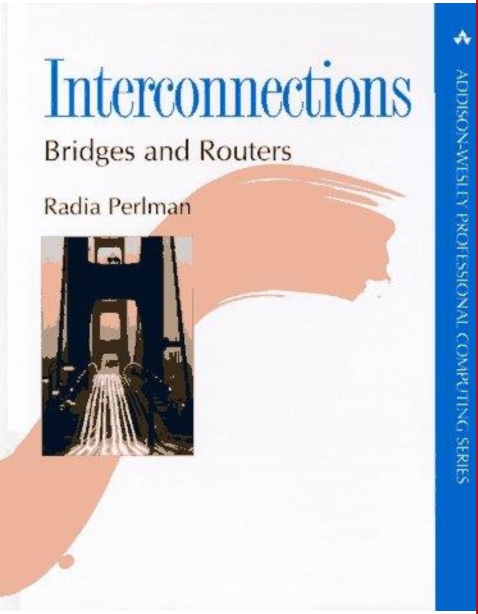
- Adjacencies are established between all neighboring routers. There is no DR or BDR for a point-to-multipoint network. No network LSA is originated for point-to-multipoint networks. Router priority is not configured for point-to-multipoint interfaces or for neighbors on point-to-multipoint networks.
- When originating a router LSA, the point-to-multipoint interface is reported as a collection of point-to-point links to all the adjacent neighbors on the interface. This is together with a single stub link advertising the IP address of the interface with a cost of 0.

Configuration

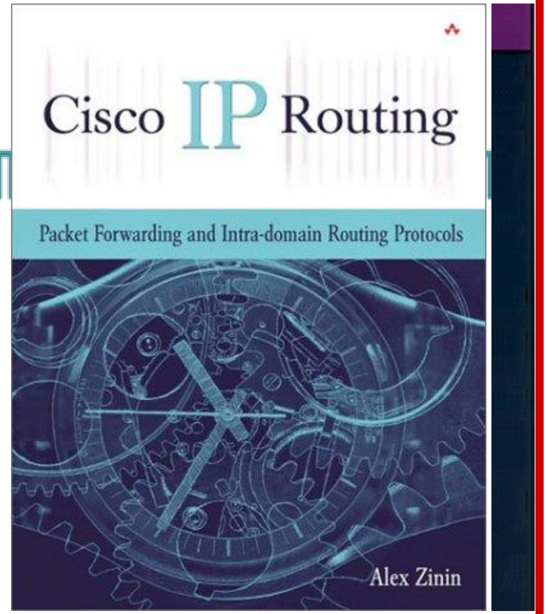
Cisco - Router

```
<Configuration for RTA>
!
interface Serial0
   encapsulation frame-relay
   ip address 3.1.1.1 255.255.255.0
   ip ospf network point-to-multipoint
   frame-relay map ip 3.1.1.2 22 broadcast
!
router ospf 1
   network 3.1.1.0 0.0.0.255 area 0
---------------------------------------------
<Configuration for RTB>
!
interface Serial0
   encapsulation frame-relay
   ip address 3.1.1.2 255.255.255.0
   ip ospf network point-to-multipoint
   frame-relay map ip 3.1.1.1 200 broadcast
   frame-relay map ip 3.1.1.3 300 broadcast
!
router ospf 1
   network 3.1.1.0 0.0.0.255 area 0
---------------------------------------------
<Configuration for RTC>
!
interface Serial0
   encapsulation frame-relay
   ip address 3.1.1.3 255.255.255.0
   ip ospf network point-to-multipoint
   frame-realy map ip 3.1.1.3 33 broadcast
!
router ospf 1
   network 3.1.1.0 0.0.0.255 area 0
```

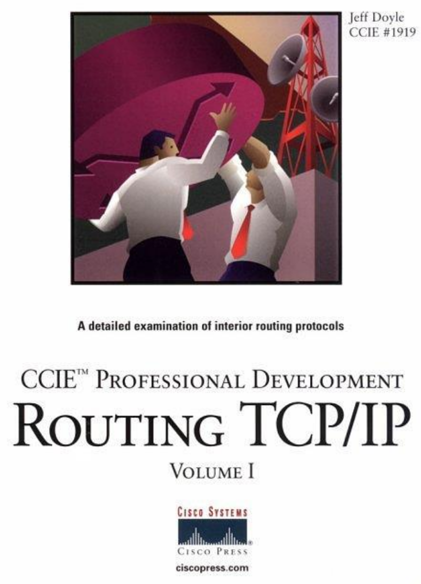
Next week… Multi Area OSPF



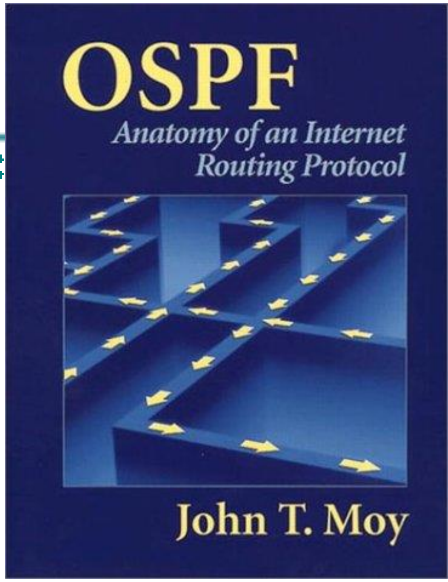
Interconnections : Bridges and Routers by Radia Perlman



Cisco IP Routing: Packet Forwarding & Intra-domain Routing Protocols by Alex Zinin



Routing TCP/IP Volume I by Jeff Doyle



OSPF, Anatomy of an Internet Routing Protocol by John Moy (creator of OSPF)

- **For more information on OSPF, link-state routing protocol, Dijkstra's algorithm and routing in general, check out these sources.**